Multivariate and Naïve Bayes Text Classification Approach to Cost Growth Risk in Department of Defense Acquisition Programs

THESIS

Charlton E. Freeman, Captain, USAF

AFIT-ENC-13-M-03

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT-ENC-13-M-03

Multivariate and Naïve Bayes Text Classification Approach to Cost Growth Risk In Department of Defense Acquisition Programs

THESIS

Presented to the Faculty

Department of Mathematics and Statistics

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Cost Analysis

Charlton E. Freeman, BBA

Captain, USAF

March 2013

AFIT-ENC-13-M-03

Multivariate and Naïve Bayes Text Classification Approach to Cost Growth Risk in Department of Defense Acquisition Programs

Charlton E. Freeman, BBA
Captain, USAF

Approved:

_____	_____
Edward D. White, Ph.D (Chairman)	Date
_____	_____
Jonathan D Ritschel, LtCol, USAF (Member)	Date
_____	_____
Austin W. Dowling, Capt, USAF (Member)	Date

Abstract

In these fiscally austere times, researchers have diligently sought methods to detect cost risk in the DOD acquisition programs. Our research effort reflects a culmination of three years of research seeking solutions to the problem of identifying programs with elevated levels of cost risk. Specifically, we applied multivariate classification and multinomial Naïve Bayes text classification techniques to develop three cost risk identification models. We find our model considering a 6-month change in the estimate at complete (EAC) of greater than 5% in magnitude, identified 69.5% of the high-risk programs in our dataset with 76.21% accuracy. Next, our model considering a 6-month increase in the EAC of greater than 5% correctly identified 67.90% of the high-risk programs with 79.68% accuracy. Finally, our model considering a 12-month increase in the EAC of greater than 5%, identified 91.69% of the high-risk programs with an accuracy of 78.31%. This research effort acts as a capstone, concentrating the knowledge collected from previous efforts and provides an actionable decision support tool for the DOD acquisition community. We find this research directly supports the goals of "more disciplined use of resources" and "improving efficiency" laid out in the OUSD(Comptroller) FY2013 Defense Budget (Department of Defense, 2012a:3.1).

I dedicate this work to my beautiful wife, for her endless patience and support. Without her pushing me forward, the title page would be my first and only page.

Acknowledgments

I would like to thank Dr. Edward White, for his guidance throughout this effort. His willingness to respond to emails at all hours of the day or night and share his wealth of wisdom proved instrumental. I would also like to thank Lieutenant Colonel Dan Ritschel, for listening to my seemingly endless rambling and providing his sound insights and expertise. Finally, I would like to thank Captain Austin Dowling, for providing his unique contributions and patience while I barraged him with questions.

Charlton E. Freeman

Table of Contents

Page

Page

Page

List of Figures

Page

Page

Page

Page

List of Tables

Page

Page

List of Equations

Page

Page

Multivariate and Naïve Bayes Text Classification Approach to Cost Growth Risk in Department of Defense Acquisition Programs

I. Introduction

After a decade of war, the Department of Defense (DOD) began the process of realigning priorities and budgets to reflect the drawdowns in Iraq and Afghanistan. Additionally, the DOD must deal with the added pressures of the 2011 Budget Control Act's requirement to reduce expenditures by $259 billion over the next five years (Department of Defense, 2012b). All levels of Defense financial management face tight budgets, highly scrutinized expenditures, and greater accountability.

Thirty percent of the $678.7 billion DOD budget request for Fiscal Year 2012 consisted of acquisition costs (Office of the Under Secretary of Defense (Comptroller), 2011). The success or failure of the acquisition enterprise depends on the careful management of cost. To that end, prior research (Keaton, White, & Unger, 2011; Dowling, 2012; Miller, 2012; Dowling, Miller, & White, 2012) sought the development of methods to forecast changes in the Estimate at Complete (EAC) for acquisition programs. This early identification draws the Program Manager's attention to areas that have the potential to become costly issues. Keaton et al. (2011) and Dowling et al. (2012) focused on the application of Statistical Process Control (SPC) methods to Earned Value Management (EVM) data to identify programs with high-risks of cost growth. While other methods exist for measuring cost growth, here we measure cost growth by changes in the EAC of Major Defense Acquisition Programs (MDAPs) (Hough, 1992:10-11). The

previous works showed a promising start to the idea of detecting elevated levels of cost growth risks in EVM data but suffered from somewhat low probabilities of an issue actually occurring given the model identified the program as at risk for cost growth. Dowling et al. (2012) showed a 0.53 probability of a monthly change in the EAC of greater than 5% in magnitude occurring within six months given their model identified the program as at risk for cost growth. To date, this is the highest probability achieved using SPC methods but Program Managers must have higher certainty an issue will occur if they are to take action based on these model outputs.

Through this research, we investigate alternative techniques to improve the detection of potential cost growth. We introduce analysis of MDAPs through the application of multivariate classification methods and a multinomial Naïve Bayes text classification model to EVM data. The results of this research provide Program Managers an alternative method to differentiate between programs with nominal cost growth risk and those with high-risks of cost growth within MDAPs with a higher probability of success. Specifically, this research effort sought to answer the following questions:

1. Does adopting either a multivariate classification, multinomial Naïve Bayes text classifier, or a hybrid of the two methods, improve on prior methods used to identify programs at risk of a 6-month change in the EAC?
2. If so, do these new methods allow us to identify programs at risk of cost growth greater than 5% 6-months out? 12-months out?

3. If we answer questions one and two affirmatively, can we incorporate these methods into tools available to the DOD program management community?

The remainder of this thesis proceeds as follows: First, in Chapter II we conceptualize the application of EVM and the use of EVM data to identify high-risk programs, specifically how prior research addressed this issue. We then provide an overview of multivariate classification and multinomial Naïve Bayes text classification. Next, in Chapter III we describe the application of these methods in this study, and present the results of our application of these methods in Chapter IV. This thesis concludes with Chapter V, where we summarize the contributions and limitations of this study as well as provide direction for future research.

II. Literature Review

The DOD has struggled with cost overruns for decades (Calcutt, 1994; Sullivan, 2001). The effect of even small changes in the EAC of an ACAT I program can ripple throughout the entire DOD acquisition portfolio. Each year the Government Accountability Office (GAO) produces a report outlining the performance of the DOD Acquisition portfolio. The estimated value for all DOD acquisitions stands at $1.58 trillion. In 2011, the DOD acquisition portfolio experienced a 5% cost growth. This seemingly small number equated to a $74.4 billion increase to the expected cost of these weapon systems. These increases led to a loss in purchasing power, and a decrease in the number of programs in the acquisition portfolio (United States GAO, 2012).

This research effort focuses on providing decision makers with a decision support tool that accurately identifies high-risk programs early in the acquisition process with low false detections rates as well as low failure-to-detect rates. This early warning allows DOD decision makers and Program Managers the opportunity to apply their expertise to mitigate or even avoid potentially costly issues that could go undetected until too late. We show we can accomplish this using our alternative detection methods.

This chapter reviews current literature on the application of EVM in the DOD acquisition environment, the current research seeking the detection of high-risk programs, and concludes with an introduction to the literature supporting our proposed alternative risk detection methods.

Earned Value Management Overview

Beginning in the 1960s, the DOD implemented EVM to monitor technical, cost, and schedule performance of acquisition programs (Kwak, 2012). This tool supports Program Managers by providing them vital information on the overall health of the acquisition program as well as the ability to anticipate future issues. EVM supports the Program Manager through three main elements. First, Program Managers create a project plan/schedule that explains what work to accomplish and when. Second, EVM reports the actual cost of work performed. Third, EVM establishes the rules and metrics designed to quantify completed work on the project (Air Force Cost Analysis Agency, 2007:13.1). These three elements allow the Program Managers to track the progress of the acquisition programs and manage very complex systems. Unless otherwise noted, the remainder of this discussion focuses on material found in the Air Force Cost Analysis Handbook (Air Force Cost Analysis Agency, 2007:13.1-13.78), and highlights common terms and methods used in EVM.

EVM Terminology

EVM is a complex system that uses specialized terms to describe specific elements that relate to cost, schedule, and their derivatives. Table 1 provides a few key terms and acronyms that we use throughout our analysis. In the next section, we turn our attention to the Contractor Performance Report (CPR).

Table 1. Key Terms (Air Force Cost Analysis Agency, 2007:13.10-13.11)

Term	Description
BAC – Budget at Complete	Total budget for the total contract through any given work level
PMB - Performance Measurement Baseline	Time phased approved program budget for the contract
BCWS - Budgeted Cost for Work Scheduled	Value of work planned to be accomplished (also called Planned Value (PV))
BCWP - Budgeted Cost for Work Performed	Value of work actually accomplished (also called Earned Value (EV))
ACWP - Actual Cost of Work Performed	Cost of work actually accomplished (also called Actual Cost (AC))
SV - Schedule Variance	Difference between planned and actual schedule accomplishment
CV - Cost Variance	Difference between planned and actual cost accomplishment
CPI – Cost Performance Index	Ratio of BCWP to ACWP; measure of cost efficiency
SPI – Schedule Performance Index	Ratio of BCWP to BCWS; measure of schedule efficiency
EAC - Estimate at Completion	Estimate of total cost at completion (through any work level of the contract)
LRE – Latest Revised Estimate	Contractor's EAC

Contractor Performance Reports

The CPR provides us with many of the values for terms discussed in Table 1. The CPR acts as the main method to document cost and schedule data from the contractors (Defense Acquisition University, 2012). The CPR consists of five formats; we describe these in Table 2. These Formats provide data on contract performance, which contribute significantly to the research discussed here.

We recognize as of 1 July 2012, DOD has combined the CPR and Integrated Master Schedule into an Integrated Program Management Report (IPMR). This does not affect our analysis since the first five formats of the IPMR mirror the CPR data. Going forward any references to the CPR is interchangeable with the first five formats of the IPMR (Defense Acquisition University, 2012).

Table 2. CPR Format Descriptions (Air Force Cost Analysis Agency, 2007:13.29)

Format Title	Frequency	Description	Use of Format
1. Work Breakdown Structure (WBS)	Monthly or weekly basis as provided in contract	Report WBS element performance data (BCWS, BCWP and ACWP) for the current reporting month as well as cumulative to date data. Cost and schedule variance are calculated and reported. Identifies any reprogramming adjustment, budget at completion, and/or estimate	Isolate key cost and schedule variances, quantify the impact, analyze and project future performance. Performance issues isolated at lowest level and analyzed for impact to overall cost and schedule variances.
2. Organization Categories	Monthly or weekly basis as provided in contract	Reports the same data as Format 1 but identified by contractor functional labor categories, major subcontractors, and material. Data is summarized for the total program at the contract level.	Isolate performance issues to the contractors functional organization by major subcontractors or by material. This allows analysis and problem isolation to either internal or external areas which enables the contractor to determine the impact to overall cost and schedule of the program.
5. Explanation and Problem Analyses	Monthly	Narrative explanation of key cost, schedule, and associated variances. Contractor identifies program impacts, corrective action plans, and analyses significant drivers at the lowest specified level and at the total contract level. Includes analysis of Management Reserve and overall risk.	Correlated with data from Format 1 and 2 to understand reasons for the variances. Understanding the underlying reasons and the contractors get well plans help the analyst to prepare an integrated assessment of past and future trends and analysis overall. PM can then make informed decisions.

EVM analysis techniques

"The CPR's primary value to the government is its ability to reflect current contract status and reasonably project future program performance" (Defense Cost and Resource Center, 2005). Analysts use the CPR data to conduct investigations into the contract status and program performance. Table 3 provides common formulas used by analysts followed by a discussion on key formula uses.

Table 3. EVM Variance and Index Formulae (Air Force Cost Analysis Agency, 2007:13.38)

Variances

Favorable is Positive, Unfavorable is Negative

Cost Variance	**CV**	**= BCWP – ACWP**	**CV%**	**= (CV / BCWP) * 100**
Schedule Variance	**SV**	**= BCWP – BCWS**	**SV %**	**= (SV / BCWS) * 100**
Variance at Completion	**VAC**	**= BAC – EAC**		

Performance Indices

Favorable is > 1.0, Unfavorable is < 1.0

Cost Efficiency	**CPI**	**= BCWP / ACWP**
Schedule Efficiency	**SPI**	**= BCWP / BCWS**

Overall Status

% Schedule	**= ($BCWS_{CUM}$ / BAC) * 100**
% Complete	**= ($BCWP_{CUM}$ / BAC) * 100**
% Spent	**= ($ACWP_{CUM}$ / BAC) * 100**

Estimate at Completion*

EAC	**= Actuals to Date + [(Remaining Work) / (Efficiency Factor)]**
EAC_{CPI}	**= $ACWP_{CUM}$ + [(BAC – $BCWP_{CUM}$) / CPI_{CUM}] = BAC / CPI_{CUM}**
$EAC_{Composite}$	**= $ACWP_{CUM}$ + [(BAC – $BCWP_{CUM}$) / (CPI_{CUM} * SPI_{CUM})]**

To Complete Performance Index (TCPI)**

$TCPI_{EAC}$	**= Work Remaining / Cost Remaining = (BAC – $BCWP_{CUM}$) / (EAC – $ACWP_{CUM}$)**

* To determine a contract level TCPI or EAC replace BAC with TAB

** To determine the TCPI_BAC,LRE replace EAC with EITHER BAC or LRE

Analysts regularly use the formulas in Table 3 to establish the health of acquisition programs. A few metrics relevant to this study include SV, CV, CPI, SPI, and EAC. The following paragraphs will provide a clearer understanding of these terms.

SV provides the analyst with a performance measure with respect to the PMB. Favorable SV, characterized by positive values, indicates the program progressing ahead

of schedule, while a negative value, or unfavorable SV, indicates the program lags behind schedule. We find SV percent useful when comparing multiple programs as it negates the dissimilarities in program funding scale and allows a meaningful direct comparison.

CV identifies the differences between the budgeted cost of work accomplished and the actual cost. Favorable CV, also evidenced by a positive value, indicates a potential surplus of funding, while an unfavorable CV indicates the program has the potential for a budget overrun. Similar to SV percent, the CV percent also negates the dissimilarities in program funding scale between programs.

The CPI indicates the cost efficiency of a project. We accomplish this by showing a ratio of dollars budgeted versus dollars spent. A program with a CPI of 1.0 indicates the program earns as many budgeted dollars as it spends. A CPI less than 1.0 shows the program spending in excess of the budgeted amount. A CPI above 1.0 indicates a program spends less money than the budgeted amount.

The SPI measures the schedule efficiency of an acquisition program. A favorable SPI of greater than 1.0 show the program earning credit for more work than scheduled, or ahead of schedule. Conversely, an unfavorable SPI of less than 1.0 indicates the program earning credit for less work than scheduled, or behind schedule.

EAC provides an estimate of the total cost of a program. Analysts typically see two EACs. First, there is a Government estimate of the project. Secondly, the Defense contractors provide an EAC (usually three: worst case, best case, and most likely case), also known as the Latest Revised Estimate (LRE) (Air Force Cost Analysis Agency, 2007). In this analysis, we use LRE and EAC interchangeably and find Defense

contractors generally provide three EAC estimates in the Format 1s. The contractors provide their EAC-worst case, EAC-best case, and EAC-most likely. We use all three in our analysis but we focus on the EAC-most likely to observe changes; we simply refer to the EAC-most likely as EAC. The EAC provides Program Managers with a good crosscheck for identifying potential cost increase at different levels within the program.

Recent research has expanded the tools available to Program Managers and analysts for identifying programs at risk of a change in the EAC within 6-months through the analysis of EVM data (Keaton, White, & Unger, 2011; Dowling, 2012; Miller, 2012; Dowling, Miller, & White, 2012). The following discussion provides a review of the current literature on these techniques.

Increased Risk detection in EVM

Risk defined

As previously discussed, the EAC provides the Program Manager with the anticipated costs of the completed program. Each month, the program's efficiency may change and this change produces a higher or lower EAC. Keaton, White, & Unger (2011) pioneered the application of SPC methods to predict major changes to the EAC using Statistical Process Control methods. They defined a major change in the EAC as a monthly change greater than 5% in magnitude. Smaller changes occur regularly and did not raise concerns in their analysis. Later research in predicting major changes to the EAC (Dowling, 2012; Dowling, Miller, & White, 2012; Miller, 2012) continued the use of this definition when identifying increased risk in an Acquisition program. Originally,

the authors used the term *problem* in place of *risk*. According to the Risk Management Guide for DOD Acquisition (OUSD(AT&L), 2006:1), "Risk is a measure of future uncertainties in achieving program performance goals and objectives within defined cost, schedule and performance constraints." The Risk Management Guide for DOD Acquisition (OUSD(AT&L), 2006:1) goes on to describe issues, or problems, as events that have already happened with certainty. We then argue that prior research (Dowling, 2012; Dowling, Miller, & White, 2012; Miller, 2012; Keaton, White, & Unger, 2011) focused on evaluating historical data to identify future programs with elevated levels of risk and uncertainty associated with cost growth, as measured by the EAC, and not on identifying problems that have already occurred.

Increased Risk Detection Methods

Over the last two years, we have seen an increased interest in research seeking improved methods to anticipate major changes in the EAC (Dowling, 2012; Dowling, Miller, & White, 2012; Miller, 2012). We see each effort explored different aspects of available data but all prior research focused on the use of Statistical Process Control to identify what programs would suffer from a monthly change in the EAC greater than 5% in magnitude within a specified timeframe (see Table 4).

Keaton et al. (2011) sought to develop a model focused on predicting programs at risk of a change in the month over month EAC greater than 5% in magnitude. Keaton et al. accomplished this by using data from the contract history files and Autoregressive/Integrated/Moving Average to identify statistical differences to monitor changes in the CPI and SPI. Finally, they applied Statistical Process Control (SPC) to

identify programs expected to experience a monthly change greater than 5% in magnitude.

Dowling (2012) developed an optimization model that attempted to predict the future EAC of a program. He then compared this predicted EAC with the current month EAC to create an EAC ratio input for his SPC control bounds. These control bounds provided the mechanism to identify programs at risk of a monthly change greater than 5% in magnitude within a specified period.

Similarly, Miller (2012) applied Latent Dirichlet Allocation (LDA) text mining methods to analyze the Format 5s and produced inputs used in an Ordinary Least Squares (OLS) regression model to predict a future EAC. He then compared the predicted EAC value of the model against the actual values of the EAC. This ratio served as inputs to the SPC model, which again served as the mechanism to identify programs at risk of a monthly change greater than 5% in magnitude with a specified period.

Dowling et al. (2012) developed a weighted average from the outputs of Dowling (2012) and Miller (2012) to produce a model considering both Format 1 and Format 5 data. This weighted average served as the inputs for their SPC Model. This linking of the two models produced an overall improvement over the outputs of each model independent of the other.

Table 4. Risk Identification Methods

Authors	Variables	Data Source	Detection Method
Keaton, White, & Unger (2011)	CPI, SPI	Contract history files	Statistical Process Control
Dowling (2012)	148+ variables (variations and combinations of data found on Format 1)	Contractor Performance Report – Format 1	Statistical Process Control
Miller (2012)	Text	Contractor Performance Report – Format 5	Statistical Process Control
Dowling, Miller, & White (2012)	148+ variable & text	Contractor Performance Report – Format 1 & 5	Statistical Process Control

SPC provides process managers a statistical tool that identifies quality control problems. Typically, SPC attempts to measure characteristics of a product as it moves through the manufacturing process. When a product's characteristics fall outside some predetermined upper and lower acceptable boundaries, or limits, we identify this process as out of control and require adjustments to bring the product's characteristics back within acceptable ranges. We see this specific tool of SPC referred to as a Control Chart (Thompson & Koronacki, 2002:53-71).

We see from the Program Manager's perspective that with an in control process we expect the EAC to remain somewhat constant. If conditions, as indicated by the variables observed in Table 3, begin to deteriorate or substantially improve, we expect the EAC to increase or decrease respectively. Given a limited amount of information and time, the Program Manager would find it beneficial to identify these areas of concern and focus their time and talents on these high-risk areas.

Control Chart Effectiveness

Any tool used to support decisions must provide reliable information to a decision maker. In the context of Control Charts, we expect to find high levels of certainty that if the acquisition program's measured characteristics fall within the acceptable range the program will not experience cost growth. Conversely, if the program's measured characteristics fall outside the acceptable ranges, we expect a high level of certainty the program will experience cost growth. Prior research has shown promising results in these areas.

Figure 1 depicts a desired classification matrix for any risk identification method. Figure 2 shows the conditional probability outcomes of the prior works focusing on the six month timeframe (Keaton, White, & Unger 2011; Dowling, 2012; Dowling, Miller, & White, 2012; Miller, 2012). The principle diagonal of the classification matrices details the correctly identified observations in the analysis; in Figure 1, we identify the high probability elements as the principle diagonal. The off diagonal of the classification matrices details the incorrectly identified observations in the analysis; in Figure 1, we identify the low probability elements as the off diagonal. These results correspond to a 6-month detection window. Meaning if the model identifies a program as high-risk, the authors counted the detection correct if the EAC experienced a monthly change of greater than 5% in magnitude within 6-months.

Desired Classification Matrix			
		Predicted Class	
		High-Risk	Nominal Risk
Actual Class	High-Risk	High Probability	Low Probability
	Nominal Risk	Low Probability	High Probability

Figure 1. Desired Classification matrix

Keaton, White, & Unger (2011)			
		Predicted Class	
		High-Risk	Nominal Risk
Actual Class	High-Risk	0.2269	0.2800
	Nominal Risk	0.7731	0.7200

Dowling (2012)			
		Predicted Class	
		High-Risk	Nominal Risk
Actual Class	High-Risk	0.4236	0.1798
	Nominal Risk	0.5764	0.8202

Miller (2012)			
		Predicted Class	
		High-Risk	Nominal Risk
Actual Class	High-Risk	0.3988	0.2017
	Nominal Risk	0.6012	0.7983

Dowling, Miller, & White (2012)			
		Predicted Class	
		High-Risk	Nominal Risk
Actual Class	High-Risk	0.5290	0.1831
	Nominal Risk	0.4710	0.8169

Figure 2. Detection Comparison (Dowling, Miller, & White, 2012)

Alternative Detection Method: Multivariate Classification

From Figure 2, we see a sharp improvement in the ability to anticipate major EAC changes. Initial efforts showed the probability of successfully identifying the high-risk programs, or programs expected to experience a monthly cost growth of greater than 5% in magnitude within six months, at 0.227. The most recent efforts improved the probability of a successful high-risk detection to 0.529. We see this as an opportunity to submit Discrimination and Classification as an alternative detection method and further increase the detection rate.

Discrimination seeks to separate groups of data as much as possible (Johnson & Wichern, 2007:575). In comparison, classification seeks to create a rule that allows the accurate assignment of new observations to a particular group. The goals of

discrimination and classification tend to overlap and often accomplished simultaneously. Going forward we simply refer to discrimination and classification as classification.

In the context of EVM risk identification, we have two classes: high-risk programs and nominal risk programs. Through the analysis of historical monthly CPR data, we know which programs belong to a given class. We use classification methods to analyze historical data and create classification rules that will properly assign a new observation to the high-risk or nominal risk class. A good classification rule will result in few misclassifications. In other words, a good classification rule would mirror the results from the desired classification matrix in Figure 1.

Additionally, comprehensive classification models takes into account prior probabilities. These prior probabilities incorporate already understood information about a population of interest into a model. For example, "if we really believe that the (prior) probability of a financially distressed and ultimately bankrupted firm is very small, then one should classify a randomly selected firm as nonbankrupt unless the data overwhelmingly favors bankruptcy" (Johnson & Wichern, 2007:578). If there are no assumptions made for the prior probability of each class, we can leave the probability of each class as equally likely (or 0.5).

Next, we consider the cost of misclassification, another important aspect of classification (Johnson & Wichern, 2007:581). Where possible, a good classification model accounts for the cost of misclassifying an observation. In the case of EVM risk detection, we find it difficult to attribute specific costs of misclassification. For example, we do not have clear accounting of costs associated with falsely identifying programs as

high-risk. We understand the additional cost in the form of person-hours or additional resources used to identify and mitigate the root cause of risks that never materializes can become substantial but these costs are unknown. In the absence of clear information, we can assume (however unlikely) these costs to be equal. Figure 3 represents the costs of misclassification depicted by a cost matrix, where c(2|1) represents the cost of misclassifying a high-risk program as a nominal risk program and c(1|2) depicts the cost of misclassifying a program with nominal risk as high-risk.

		Classify as:	
		High-Risk	Nominal Risk
True Population	High-Risk	0	c(2|1)
	Nominal Risk	c(1|2)	0

Figure 3. Misclassification Cost Matrix (Johnson & Wichern, 2007:581)

In classification analysis, we use statistical principles to describe the characteristics of each class. This description of each class results in a probability density function for each class. The normal distribution provides an example of a well-known probability density function. For now, let $f_1(\boldsymbol{x})$ and $f_2(\boldsymbol{x})$ represent the probability density functions of the high-risk class and nominal risk classes respectively.

We tie all these classification concepts together in a discussion about the minimization of Expected Cost of Misclassification (ECM) (Johnson & Wichern, 2007:581). As the name implies, the ECM provides the expected cost of misclassifying observations. We calculate the ECM by multiplying the off-diagonal entries in Figure 3 by the probabilities of occurrence, defined in Equation 1 as p_1 for the prior probability of

high-risk program class and p_2 for the prior probability of nominal risk program class. ECM also considers the probability of misclassifying an observation, here defined as $P(2|1)$ if we identify the program as nominal risk if it truly belongs to the high-risk program and $P(1|2)$ if we misclassify a nominal risk program as high-risk. Classification rules should seek to minimize the ECM. Equation 2 and Equation 3 define the classification regions R_1 and R_2 that minimize the ECM.

$$ECM = c(2|1)P(2|1)p_1 + c(1|2)P(1|2)p_2 \tag{1}$$

$$R_1: \frac{f_1(\boldsymbol{x})}{f_2(\boldsymbol{x})} \geq \left(\frac{c(1|2)}{c(2|1)}\right)\left(\frac{p_1}{p_2}\right) \tag{2}$$

$$(Density\ Ratio) \geq (CostRatio)(Prior\ Probability\ Ratio)$$

$$R_2: \frac{f_1(\boldsymbol{x})}{f_2(\boldsymbol{x})} < \left(\frac{c(1|2)}{c(2|1)}\right)\left(\frac{p_1}{p_2}\right) \tag{3}$$

$$(Density\ Ratio) < (CostRatio)(Prior\ Probability\ Ratio)$$

In our discussion of high-risk program detection in EVM, we can see that given some density functions, $f_1(\boldsymbol{x})$ and $f_2(\boldsymbol{x})$, we have the ability to determine which class to assign the observation. If the density ratio is greater than or equal to the cost ratio multiplied by the prior probability ratio, we assign the observation to the high-risk class, as seen in Equation 2. We then assign observations that are less than these ratios to the nominal risk class, as shown in Equation 3. In Chapter III, we further discuss how we

determine the density function, what characteristics define the model, and how to apply classification analysis to EVM data to provide an alternative problem detection method.

Alternative Detection Method: Multinomial Naïve Bayes Classifier

Miller (2012) used text-mining analysis, specifically, Latent Dirichlet Allocation Self-Organizing Map, to identify programs at high-risk of cost growth. From Figure 2, we see text analysis on the Format 5s showed the potential to differentiate the two classes but suffered from low probabilities a program would incur cost growth given the model identified the program as high-risk. In this section, we introduce an alternative method to identify programs with high-risk of cost growth through the analysis of Format 5 data, the multinomial Naïve Bayes classifier.

Manning, Raghavan, & Schutze (2008:236-237) introduce the concept of multinomial Naïve Bayes Classifier within machine learning-based text classification. Machine learning automatically constructs the criteria for class assignment by learning the class characteristics from the training data (the dataset less the validation set). We focus in on a specific type of learning in our research, supervised learning. In supervised learning, the researchers provide manually labeled observations to the classifier for learning. We use the training set to train the model then apply the finalized model to the validation set for a final measure of the expected performance of the model on new data. In our research effort, we differentiate high-risk programs from nominal risk programs using multinomial Naïve Bayes model described here.

Manning et al. (2008) characterize multinomial Naïve Bayes model as a probabilistic learning model and defined in Equation 4:

$$P(c|d) \propto P(c) \prod_{1 \leq k \leq n_d} P(t_k|c) \tag{4}$$

where *P(c|d)* is the conditional probability of class *c* given document *d*, *P(c)* is the prior probability of class *c*, $P(t_k|c)$ is the conditional probability of term t_k in the document *d* given class *c*, n_d represents the total number of tokens, or words, considered in the document (Manning, Raghavan, & Schutze, 2008:239). $P(c|d)$ is proportional to $P(c) \prod_{1 \leq k \leq n_d} P(t_k|c)$ because we have dropped *P(d)* from the denominator of Bayes' rule, $P(c|d) = \frac{P(c) \prod_{1 \leq k \leq n_d} P(t_k|c)}{P(d)}$. Later, we compare the probabilities between $P(High - risk|d)$ and $P(Nominal\ risk|d)$. During this comparison $P(d)$ remains constant; therefore, we set $P(c|d)$ proportional to $P(c) \prod_{1 \leq k \leq n_d} P(t_k|c)$. To clarify Equation 4 further consider the following excerpt from Manning, Raghavan, & Schutze (2008):

> $\langle t_1, t_2, \ldots t_{n_d} \rangle$ are the tokens in *d* that are part of the vocabulary we use for classification and n_d is the number of such tokens in *d.* For example, $\langle t_1, t_2, \ldots t_{n_d} \rangle$ for the one-sentence document *Beijing and Taipei join the WTO* might be $\langle Beijing, Taipei, join, WTO \rangle$, with $n_d = 4$ if we treat the terms *and* and *the* as stop words (Manning, Raghavan, & Schutze, 2008:239).

We define Stop words as extremely common words that provide insignificant information when differentiating between classes (Manning, Raghavan, & Schutze, 2008:25). We completely exclude stop words from the analysis vocabulary. With the probabilistic learning model defined, we turn our attention to development of the decision criteria used to assign a class to a document.

To meet the objective of identifying the best class of a document we look to the most likely class or *maximum a posteriori* (MAP) class c_{map} displayed here in Equation 5 (Manning, Raghavan, & Schutze, 2008:239).

$$c_{map} = \underset{c \in \mathbb{C}}{argmax}\ \hat{P}(c|d) = \underset{c \in \mathbb{C}}{argmax}\ \hat{P}(c) \prod_{1 \le k \le n_d} \hat{P}(t_k|c) \tag{5}$$

where $\hat{P}(c)$ is defined as the probability estimate of $P(c)$ using data in the training set, $\hat{P}(t_k|c)$ is the estimated conditional probability of word t_k in class *c*, and $\mathbb{C}$ is a fixed set of classes. When evaluating Equation 5, multiplying a large number of conditional probabilities can quickly result in a floating point underflow. Floating point underflow occurs when the number being calculated is smaller than the minimum value the computer is able to represent. Therefore, the computer represents the number as zero. To overcome this problem, Manning, Raghavan, & Schutze (2008:239) suggest computing the c_{map} using properties of logarithms. We know $\log(xy) = \log(x) + \log(y)$; therefore, when we apply this logarithm property to Equation 5, we still find the class with the higher probability as the most probable and results in Equation 6.

$$c_{map} = \underset{c \in \mathbb{C}}{argmax} \left[\log \hat{P}(c) + \sum_{1 \le k \le n_d} \log \hat{P}(t_k|c) \right] \tag{6}$$

According to Manning, Raghavan, & Schutze (2008:239) the sum of $\log \hat{P}(c)$ and $\log \hat{P}(t_k|c)$ measures the evidence the document being observed belongs to class *c*.

To estimate the parameters $\hat{P}(c)$ and $\hat{P}(t_k|c)$, we initially use the maximum likelihood estimate (MLE) (Manning, Raghavan, & Schutze, 2008:240) From Equation 6, we define the MLE of $\hat{P}(c)$ as:

$$\hat{P}(c) = \frac{N_c}{N}, \tag{7}$$

where N_c is the number of documents belonging to class *c* and *N* is the total number of documents analyzed. Additionally, in Equation 6, we define the MLE of $\hat{P}(t|c)$ as:

$$\hat{P}(t|c) = \frac{T_{ct}}{\sum_{t' \in V} T_{ct'}}, \tag{8}$$

where T_{ct} is the number of times the word *t* appears in the training document from class *c.* Strictly using MLE results in an estimate of zero for word-class combinations unseen in the training data (Manning, Raghavan, & Schutze, 2008:240). The training data is inadequate to observe every word-class combination possible or rare words, a problem commonly referred to as sparseness. Laplace smoothing provides us with a method to combat problems introduced by sparseness.

Laplace smoothing, or add-one smoothing, adds one to each count of T_{ct} in Equation 8. This is equivalent to a uniform Bayesian prior for each word. As we add new observations the uniform Bayesian prior is updated. Equation 9 displays the aforementioned Laplace smoothing.

$$\hat{P}(t|c) = \frac{T_{ct} + 1}{\sum_{t' \in V}(T_{ct'} + 1)} = \frac{T_{ct} + 1}{(\sum_{t' \in V} T_{ct'}) + B'}, \quad (9)$$

where $B = |V|$ is the cardinality, or number of words, of the training data (Manning, Raghavan, & Schutze, 2008:240). This has the effect of decrementing the probability of words actually seen and applying the reserved probability to the unseen observations. Equation 9 can be generalized to an add-α smoothing detailed in Equation 10.

$$\hat{P}(t|c) = \frac{T_{ct} + \alpha}{\sum_{t' \in V}(T_{ct'} + \alpha)} = \frac{T_{ct} + \alpha}{(\sum_{t' \in V} T_{ct'}) + \alpha B'} \quad (10)$$

where α corresponds to the belief in a uniform Bayesian prior distribution over the vocabulary (Manning, Raghavan, & Schutze, 2008:208).

We make two assumptions in the application of the multinomial Naive Bayes classifier (Manning, Raghavan, & Schutze, 2008:246-249). First, we assume a Naïve Bayes conditional independence. This means we assume the words are independent of each other given some class. In reality, we know that conditional independence does not typically hold in text. An example provided by (Manning, Raghavan, & Schutze, 2008:248) considers the word pair Hong and Kong for the class China. In everyday usage, these words express highly dependent behavior but we still treat them as independent. In this example, the dependent nature of the words does not influence the ability to apply the Naïve Bayes classifier.

Secondly, we assume positional independence of the words. Here, we give a word the same conditional probability regardless of the position of the word in the document. Models commonly called “bag of words models” make the positional independence

assumption (Manning, Raghavan, & Schutze, 2008:247). Neither of these assumptions holds in reality. "NB [Naïve Bayes] classifiers estimate badly, but often classify well" (Manning, Raghavan, & Schutze, 2008:249). In the Naïve Bayes classifier, we see that the highest score and not the accuracy of the probability estimate drives the classification decision.

In Figure 4, we tie the entire discussion of the multinomial Naïve Bayes classifier together using an algorithm adapted from Manning, Raghavan, & Schutze (2008:241). In Chapter III, we further clarify our vocabulary extraction methods and discuss our application of add-α smoothing.

TrainMultinomialNB($\mathbb{C}, \mathbb{D}$)
1 $V \leftarrow$ EXTRACTVOCABULARY($\mathbb{D}$)
2 $N \leftarrow$ CountDocs($\mathbb{D}$)
3 **for each** $c \in \mathbb{C}$
4 **do** $N_c \leftarrow CountDocsInClass(\mathbb{D}, c)$
5 $prior[c] \leftarrow \frac{N_c}{N}$
6 $text_c \leftarrow ConcatenateTextOfAllDocsInClass(\mathbb{D}, c)$
7 **for each** $t \in V$
8 **do** $T_{ct} \leftarrow CountOfWords(text_c, t)$
9 **for each** $t \in V$
10 **do** $condprob[t][c] \leftarrow \frac{T_{ct}+\alpha}{(\sum_{t' \in V} T_{ct'})+\alpha B'}$
11 **return** $V, prior, condprob$

ApplyMultinomialNB($\mathbb{C}, V, prior, condprob, d$)
1 $W \leftarrow ExtractWordsFromDoc(V, d)$
2 **for each** $c \in \mathbb{C}$
3 **do** $score[c] \leftarrow \log prior[\mathrm{c}]$
4 **for each** $t \in V$
5 **do** $score[c] += \log condprob[t][c]$
6 **return** $argmax_{c \in \mathbb{C}}\, score[c]$

Figure 4. Naïve Bayes algorithm (multinomial model): Training and testing adapted from Manning, Raghavan, & Schutze, (2008:241)

Summary

This chapter provided a review of current literature on the application of EVM in the DOD, the current research seeking the detection of high-risk acquisition programs using EVM data, and introduced literature supporting two alternative problem detection methods, multivariate classification, and multinomial Naïve Bayes text classification. In the next chapter, we delve deeper into the application of multivariate classification to EVM data, analysis of Format 5 data by applying the multinomial Naïve Bayes text classification model, and integration of the two methods to improve identification of high-risk acquisition programs.

III. Methodology

In this chapter, we provide a detailed description of the analysis conducted for this study. We comprise this study in three distinct components. First, we begin detailing the analysis of EVM data using Multivariate Classification techniques to identify high-risk acquisition programs. Secondly, we introduce a multinomial Naïve Bayes classification technique on the Format 5 data to identify high-risk programs. We conclude this chapter by detailing the integration of the Multivariate Classification technique and the multinomial Naïve Bayes classifier to produce a new risk detection method.

Multivariate Classification

Database

This study focuses on detecting risk in MDAPs and seeks to improve on previously developed models (Dowling, 2012; Dowling, Miller, & White, 2012; Miller, 2012). We elected to use the database collected for these previous studies in our analysis. This allows for more comparable results between this study and prior studies. Additionally, using the same database eliminates any improvements in results associated with additional data unavailable to the prior studies. Next, we provide a discussion on this database, including: target data, data collection, additional data calculations, other considerations, validation set, and limitations.

Target data

This study, like prior studies, focuses on the largest DOD acquisition programs. The Acquisition community knows these programs as Acquisition Category ID (ACAT

ID) and are defined by "Research, Development, Test and Evaluation (RDT&E) expenses of more than $365 million (Fiscal Year (FY) 2000 constant dollars) or procurement of more than $2.19 billion (Fiscal Year (FY) 2000 constant dollars)" (Defense Acquisition University, 2009). Additionally, a program can be designated an ACAT ID program if the Office of the Under Secretary of Defense for Acquisition, Technology, and Logistics identifies the program as a special interest item. ACAT ID programs are the largest programs in dollar terms and experience the highest level of scrutiny and oversight. A small percentage change in these programs results in large dollar changes, meaning these programs potentially can benefit greatly from identifying high-risk programs sooner.

Data Collection

Prior researcher has utilized the Defense Cost and Resource Center (DCARC) for data on these programs (Dowling, 2012; Dowling, Miller, & White, 2012; Miller, 2012). DCARC serves as the DOD's authoritative source for EVM data, including the monthly CPR data. The original query of DCARC used to produce the database resulted in 1303 monthly CPR data points from 37 different programs ranging in dates from September 2007 to August 2011. Figure 5 provides a histogram of monthly EACs (in millions), Table 5 details the composition of the database by service and program type.

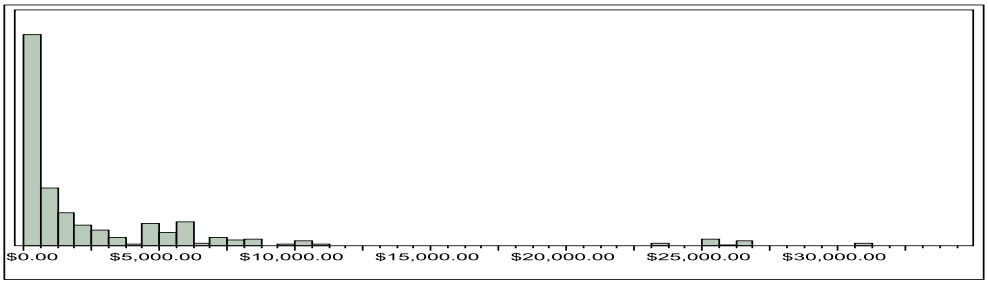


Figure 5. Database Histogram ($ Millions)

Table 5. Program Composition

Service	Quantity
AF	14
Navy	8
Army	7
Joint	7
Marine	1

Platform	Quantity
Plane	10
Comm.	9
Satellite	5
Missile	3
Helicopter	3
Radar	2
Ship	2
Facility	2
Vehicle	1

These CPRs provide us a wealth of information from the Format 1s and Format 5s. Initially, we focus strictly on the Format 1 much like Dowling (2012) but later incorporate the Format 5s using the multinomial Naïve Bayes Classifier. Table 8 shows the data fields originally collected from the Format 1s.

Table 6. DCARC Format 1 Fields

Begin Date	Report Number
Program Name	Budgeted Cost Of Work Scheduled
EAC – Best Case	Actual Cost of Work Performed
EAC – Worst Case	Budgeted Cost of Work Performed
EAC – Most likely Case	Budget At Complete

Additional Data Calculations

As previously discussed, analysts collect EVM data and perform calculations from Table 3 to understand the health of the acquisition program. By performing these calculations on our dataset, we derive the same information commonly used by the EVM analyst. Additionally, previous studies (Dowling, 2012; Dowling, Miller, & White, 2012; Miller, 2012) have included the moving three-month standard deviation for the selected variables to explain the stability of the measure used, as well as the change between the current month's observation and one to two months prior. Table 7 shows a complete list

of variables requiring additional calculations in excess of that collected from the Format 1, see Appendix A for calculation details.

Table 7. Variable List for Additional Data Calculations

6 Mo Delta	CV%	SV% StDev	CPI 2 Month Change
Prgm Name w/ Mo	% Difference Between ML and W	CV% StDev	SPI 2 Month Change
% Complete	% Difference Between ML and B	CPI 1 Month Change	TSPI 2 Month Change
CPI	% Difference Between W and B	SPI 1 Month Change	TCPI 2 Month Change
SPI	StDev CPI	TSPI 1 Month Change	SCI 2 Month Change
TSPI	StDev SPI	TCPI 1 Month Change	SV% 2 Month Change
TCPI	TSPI StDev	SCI 1 Month Change	CV% 2 Month Change
SCI	SCI StDev	SV% 1 Month Change	
SV%	TCPI StDev	CV% 1 Month Change	

Other Considerations

If we consider the change in the EAC of the program as cost growth or cost growth recovery, we know from RAND (2008:47) that system types express different levels of cost growth. Additionally, RAND (2008:73) highlighted smaller programs tend to experience higher levels of cost growth. To capture these points we have included the program type, military service, and program size in our data, see Table 8. We conceptualize small in terms of small, medium, and large programs. Small represents 33% of our data.

Table 8. Additional Variables Considered

Air Force	Facility	Radar
Army	Helicopter	Satellite
Joint	Missile	Ship
Navy	Plane	Small (< $250 million)

Since this study focuses on identifying programs at risk of cost growth in six months from the current observation, we also calculated the percent change that occurs six months from the current observation. This data populates the database of possible training data with known classes of high-risk or nominal risk. Of the original 1303 observations, the training data consisted of 1009 observations. We lose two months at the beginning of each program. We require these three months for the standard deviation calculations. Additionally, we lose six months at the end of each program. These six months allows us to calculate the 6-month cumulative change from the current observation.

Validation Set

Significant consideration was given when deciding what validation method to use. We concluded a two-part validation method would provide the most insight into the validity of our findings. We first validate our findings against a commonly used 20% randomized withhold. We achieved the 20% randomized withhold using JMP®'s random row selection and set the selection rate to 20% (SAS Institute INC, 2013a). This data was then set aside prior to any analysis or model building and provided an estimate of the performance of the model beyond the training data. There are, however, challenges associated with this validation method. Johnson & Wichern (2007:599-600) explain there are two main limitations:

(i) It requires large samples
(ii) The function evaluated is not the function of interest. Ultimately, almost all of the data must be used to construct the classification function. If not, valuable information may be lost.

We have a sufficiently large sample size to overcome the limitations associated with small sample size but we desired to minimize the concerns with the loss of information that occurs by removing a large portion of data from the training set for validation. To offset this weakness we included a second approach called Lachenbruch's "holdout" procedure. This method is commonly referred to as Leave One Out Cross Validate (LOOCV). We delve into more detail about the holdout procedure in the validation section of this chapter.

Limitations

Upon careful reflection, we find three limitations that potentially affect this database. First, we implicitly assumed each monthly CPR independent of the others. However, we know each CPR reports on trends continuing over many months throughout a programs existence. We find it unclear if this influences our random selection validation method. We attempt to overcome this limitation by applying the LOOCV method but this also has the same limitation. In our recommendations for further research, we discuss ideas to understand the influence of this limitation on this study.

The second limitation to this database resulted from the collection method. When this database was originally developed, researchers excluded certain Extensible Markup Language (XML) files due to the inability to read and interpret that specific file format. This introduces a slight selection bias because we have left out a portion of available data. We note here that DCARC has provided a CPR file viewer that should overcome this limitation with Format 1 data in future research (Defense Cost and Resource Center, 2013a).

The third limitation we find in this database relates to data gaps. This problem occurs during data collection. For example, we collect several months of data but a single observation is missing. In these cases, as with previous research (Dowling, 2012; Miller, 2012; Dowling, Miller, & White, 2012), we use linear approximation to estimate the missing observation. We calculate the linear approximate by selecting the observation immediately preceding the gap and the observation immediately following the gap. We then average these observations together and use this average in place of the missing data. These gaps are minimal, occurring only 10 times in our 1303 observations or in 0.8% of the observations. We terminate analysis of the program and treat it as if no more data is available if the gap is greater than two consecutive months. Dowling (2012:19) provided Table 9 showing the total number of programs originally collected from the DCARC database, the total number of ACAT1D programs, and the programs remaining useful after evaluation of the limitations discussed here.

Table 9. Available Data from DCARC (Dowling, 2012:19)

Category	Number of Programs
All Programs	118
ACAT 1D	64
Useable	37

Multivariate Classification Model building

In this section, we discuss the specific application of classification analysis to the database previously mentioned. We begin the discussion by describing the process of selecting a probability density function to use in this study. Next, we provide the constructs used for variable selection and model building. We conclude our discussion on

multivariate classification by outlining the decision process for model selection and validation.

In this study, we elected to evaluate the data using a multivariate normal classification model. We find support for this in Johnson & Wichern (2007:584), "classification procedures based on normal populations predominate in statistical practice because of their simplicity and reasonably high efficiency across a wide variety of population models."

Additionally, we consider what happens if the data is not multivariate normal. Again, Johnson & Wichern (2007:595) provide two options. The first option is to transform the non-normal data to data more nearly normal. Alternatively, we can apply the multivariate normal classification model without considering the parent population due to the "central limit effect" and measure the classification effectiveness.

To simplify the implementation of this model at the program level, we have elected to press forward without considering the parent population of the data. If the classification results work well and the validation set confirms the performance of the model, we find the application of the multivariate normal classification model reasonable. The multivariate normal distribution is a generalization of the univariate normal distribution and defined in Equation 11. The multivariate normal distribution is a *p*-dimensional normal distribution where $\boldsymbol{\mu}$ represent the mean of the random vector $\boldsymbol{X}$, and $\boldsymbol{\Sigma}$ represents the variance-covariance matrix of $\boldsymbol{X}$ (Johnson & Wichern, 2007:150).

$$f(\boldsymbol{x}) = \frac{1}{(2\pi)^{\frac{p}{2}}|\boldsymbol{\Sigma}|^{\frac{1}{2}}} e^{-\frac{(\boldsymbol{x}-\boldsymbol{\mu})'\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})}{2}} \tag{11}$$

This generalization of the univariate normal distribution leads us to our multivariate density ratio. In our analysis, we make no assumptions concerning the equality of covariance matrices between the high-risk and nominal risk classes. In cases were the covariance matrices are not equal between populations, we use the Quadratic Classification Rule (Johnson & Wichern, 2007:594). If during analysis the covariance matrices between the two populations equal, the quadratic classification rule simplifies to the linear classification rule. Equation 12 shows the multivariate normal density ratio and Equation 13 shows the multivariate normal density ratio simplified.

$$\frac{f_1(\boldsymbol{x})}{f_2(\boldsymbol{x})} = \frac{\left(\frac{1}{(2\pi)^{\frac{p}{2}}|\boldsymbol{\Sigma_1}|^{\frac{1}{2}}} e^{-\frac{(\boldsymbol{x}-\boldsymbol{\mu_1})'\boldsymbol{\Sigma_1}^{-1}(\boldsymbol{x}-\boldsymbol{\mu_1})}{2}}\right)}{\left(\frac{1}{(2\pi)^{\frac{p}{2}}|\boldsymbol{\Sigma_2}|^{\frac{1}{2}}} e^{-\frac{(\boldsymbol{x}-\boldsymbol{\mu_2})'\boldsymbol{\Sigma_2}^{-1}(\boldsymbol{x}-\boldsymbol{\mu_2})}{2}}\right)} \tag{12}$$

$$\frac{f_1(\boldsymbol{x})}{f_2(\boldsymbol{x})} = -\frac{1}{2}\boldsymbol{x_0}'(\boldsymbol{\Sigma}_1^{-1} - \boldsymbol{\Sigma}_2^{-1})\boldsymbol{x} + (\boldsymbol{\mu}_1'\boldsymbol{\Sigma}_1^{-1} - \boldsymbol{\mu}_2'\boldsymbol{\Sigma}_2^{-1})\boldsymbol{x_0}$$
$$-\frac{1}{2}\ln\left(\frac{|\boldsymbol{\Sigma_1}|}{|\boldsymbol{\Sigma_2}|}\right) + \frac{1}{2}(\boldsymbol{\mu}_1'\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\mu_1} - \boldsymbol{\mu}_2'\boldsymbol{\Sigma}_2^{-1}\boldsymbol{\mu_2}) \tag{13}$$

where $\boldsymbol{x_0}$ is a new observation, $\boldsymbol{\Sigma}_i$ is the covariance matrix for class *i*, and $\boldsymbol{\mu_i}$ is the mean vector for class *i.*

Equation 14 shows the classification regions using the quadratic classification rule. We replace Σ with S_i to signify the calculation of the sample covariance matrix for

class *i*. Our prior estimation of the probabilities for each class *c* is simply the maximum likelihood estimate. We calculate this using the relative frequency of each class in the training data, see Equation 7. Additionally, we lack any substantive information concerning the cost of misclassification and set these equal ($c(1|2) = c(2|1) = 1$).

$$R_1: -\frac{1}{2}\boldsymbol{x_0'}(\mathbf{S}_1^{-1} - \mathbf{S}_2^{-1})\boldsymbol{x_0} + (\boldsymbol{\mu}_1'\mathbf{S}_1^{-1} - \boldsymbol{\mu}_2'\mathbf{S}_2^{-1})\boldsymbol{x_0} - k \geq \ln\left[\left(\frac{c(1|2)}{c(2|1)}\right)\left(\frac{p_2}{p_1}\right)\right] \quad (14)$$

$$R_2: -\frac{1}{2}\boldsymbol{x_0'}(\mathbf{S}_1^{-1} - \mathbf{S}_2^{-1})\boldsymbol{x_0} + (\boldsymbol{\mu}_1'\mathbf{S}_1^{-1} - \boldsymbol{\mu}_2'\mathbf{S}_2^{-1})\boldsymbol{x_0} - k < \ln\left[\left(\frac{c(1|2)}{c(2|1)}\right)\left(\frac{p_2}{p_1}\right)\right]$$

where

$$\mathrm{k} = \frac{1}{2}\ln\left(\frac{|\boldsymbol{S_1}|}{|\boldsymbol{S_2}|}\right) + \frac{1}{2}(\boldsymbol{\mu}_1'\mathbf{S}_1^{-1}\boldsymbol{\mu_1} - \boldsymbol{\mu}_2'\mathbf{S}_2^{-1}\boldsymbol{\mu_2})$$

Variable Selection

Some variables provide useful information when building a model and others are irrelevant. To decide what variables we find relevant we elected to use forward stepwise discriminant analysis, backward stepwise discriminant analysis, and a modified random generation plus sequential selection (RGSS).

Prior to beginning any stepwise discriminant analysis, we checked for perfect correlation among variables. We found four variables perfectly correlated in our correlation matrix; Table 10 illustrates the results from our correlation analysis. We elected to remove the variables on the right of Table 10 to ensure problems associated with multicollinearity do not surface. To further understand why these correlations occur, we deconstruct the calculations of SPI and SV% in Appendix B.

Table 10. Correlation Assessment

Variable 1	Variable 2	Correlation
SPI	SV%	1
StDev SPI	SV% StDev	1
SPI 1 Month Change	SV% 1 Month Change	1
SPI 2 Month Change	SV% 2 Month Change	1

We draw on the work of Jennrich R. I. (1977a, 1977b) for an understanding of stepwise discriminant analysis. The ratio of within generalized dispersion to total generalized dispersion provides a method to determine which variable to add or delete from the model. We calculate the within generalized dispersion by taking the determinant of the within group cross-product matrix. The total generalized dispersion is the determinant of the total cross-product matrix for the variables in our analysis. Equation 15 depicts the formula described above, also known as Wilks' Λ-criterion.

$$\Lambda(\boldsymbol{x}) = \frac{|W(\boldsymbol{x})|}{|T(\boldsymbol{x})|} \tag{15}$$

Here $W(\boldsymbol{x})$ represents the within group sum of cross-products for variables $\boldsymbol{x}$ and $T(\boldsymbol{x})$ represents the total sum of cross-products for variables $\boldsymbol{x}$. R. I. Jennrich further clarifies the notation for Wilks' Λ-criterion as follows:

> Gerneralizing the W and T notation, let $\boldsymbol{u} = (u_1, \ldots, u_r)$ and $\boldsymbol{v} = (v_1, \ldots, v_s)$ be sequences of variables let $\boldsymbol{W}(\boldsymbol{u}, \boldsymbol{v})$ and $\boldsymbol{T}(\boldsymbol{u}, \boldsymbol{v})$ be the matrices whos *ij*th elements were $W(u_i, v_j)$ and $T(u_i, v_j)$, respectively. Finally, let $\boldsymbol{W}(\boldsymbol{u})$ and $\boldsymbol{T}(\boldsymbol{u})$ be abbreviated notation for $\boldsymbol{W}(\boldsymbol{u}, \boldsymbol{u})$ and $\boldsymbol{T}(\boldsymbol{u}, \boldsymbol{u})$ (Jennrich, R. I. 1977b:78).

Values for Wilks' lambda-criterion range from zero to one. Smaller values for Wilks' lambda-criterion indicate better separation between groups. Equation 16 shows the impact of adding a variable $\boldsymbol{u}$ to our variable set and we call this a partial Λ-statistic

(Jennrich, R. I. 1977b:77). We use Equation 17, an F-statistic, to test the significance of the change in $\Lambda(\boldsymbol{x})$ from adding the variable *u*, where *n* represents the total number of observations, *q* is the total number of classes, and, *p* is the number of variables currently in the analysis.

$$\Lambda(u * \boldsymbol{x}) = \frac{\Lambda\big((\mathbf{x}, \mathbf{u})\big)}{\Lambda(\boldsymbol{x})} \tag{16}$$

$$\mathrm{F} = \frac{\mathrm{n} - \mathrm{q} - \mathrm{p}}{\mathrm{q} - 1} * \frac{1 - \Lambda\big((\mathbf{x} * \mathbf{u})\big)}{\Lambda(\boldsymbol{x} * \boldsymbol{u})} \tag{17}$$

Here we take a moment to discuss the sweep operator. Jennrich, R. I. (1977a:58-62) discusses a sweep operator, or a stepwise function, that steers the selection of variables using statistical criteria. The sweep operator begins with a square matrix represented by $\boldsymbol{A} = (a_{ij})$ whose *k*th diagonal element $a_{kk} \neq 0$. If we choose to include a variable *k*, we "sweep" $\boldsymbol{A}$ on the *k*th diagonal element. This sweep results in a new matrix $\widetilde{\boldsymbol{A}} = (\tilde{a}_{\boldsymbol{ij}})$ of the same size as $\boldsymbol{A}$ given by Equation 18:

$$\begin{gathered}
\tilde{a}_{kk} = -\frac{1}{a_{kk}} \\
\tilde{a}_{ik} = \frac{a_{ik}}{a_{kk}} \\
\tilde{a}_{kj} = \frac{a_{kj}}{a_{kk}} \\
\tilde{a}_{ij} = a_{ij} - \frac{a_{ik} a_{kj}}{a_{kk}} \\
\text{for } i \neq k \text{ and } j \neq k
\end{gathered} \tag{18}$$

The sweep can be undone by performing an inverse sweep of $\boldsymbol{A}$ on the *k*th diagonal element (already in the model) outlined in Equation 19:

$$\begin{aligned} \tilde{a}_{kk} &= -\frac{1}{a_{kk}} \\ \tilde{a}_{ik} &= -\frac{a_{ik}}{a_{kk}} \\ \tilde{a}_{kj} &= -\frac{a_{kj}}{a_{kk}} \\ \tilde{a}_{ij} &= a_{ij} - \frac{a_{ik}a_{kj}}{a_{kk}} \\ \text{for } i \neq k \; and \; j \neq k & \end{aligned} \tag{19}$$

Three theorems support the sweep operator as an exchange. Here we provide the three theorems followed by a brief discussion of their application in our research. For more information on these theorems including proofs, we direct the readers to Jennrich R. I. (1977a).

> Theorem 1: Let $\boldsymbol{U}$ and $\boldsymbol{V}$ be matrices of the same size and let $\boldsymbol{A}$ be a square matrix such that
>
> $$\boldsymbol{V} = \boldsymbol{UA}$$
>
> Let $\widetilde{\boldsymbol{U}}$ be obtained from $\boldsymbol{U}$ by replacing its *k*th column by the *k*th column of $\boldsymbol{V}$ and let $\widetilde{\boldsymbol{V}}$ be obtained from $\boldsymbol{V}$ by replacing its *k*th column by minus the *k*th column of $\boldsymbol{U}$. If the *k*th diagonal element of $\boldsymbol{A}$ is nonzero and $\widetilde{\boldsymbol{A}}$ is the result of sweeping $\boldsymbol{A}$ on its *k*th diagonal element, then
>
> $$\widetilde{\boldsymbol{V}} = \widetilde{\boldsymbol{U}}\widetilde{\boldsymbol{A}}$$
>
> Theorem 2: If it is possible to sweep the partitioned matrix on the left below on each diagonal element of the square submatrix $\boldsymbol{A}_{11}$ in some order, i.e., if the required nonzero elements are encountered, then $\boldsymbol{A}_{11}$ is nonsingular and the result of the sweeping is displayed on the right:
>
> $$\begin{pmatrix} \boldsymbol{A}_{11} & \boldsymbol{A}_{12} \\ \boldsymbol{A}_{21} & \boldsymbol{A}_{22} \end{pmatrix} \rightarrow \begin{pmatrix} -\boldsymbol{A}_{11}^{-1} & \boldsymbol{A}_{\boldsymbol{11}}^{\boldsymbol{-1}}\boldsymbol{A}_{\boldsymbol{12}} \\ \boldsymbol{A}_{21}\boldsymbol{A}_{11}^{-1} & \boldsymbol{A}_{22} - \boldsymbol{A}_{21}\boldsymbol{A}_{11}^{-1}\boldsymbol{A}_{12} \end{pmatrix}$$

> Theorem 3: If $\boldsymbol{A}$ is a positive definite matrix, then its diagonal elements are nonzero and remain nonzero after any sequence of sweeps (Jennrich R. I., 1977a:60-62).

We see from Theorem 1 that performing the sweep operator is equivalent to rearranging the matrix and does not affect the equality of the matrices. Theorem 2 shows the sweeping of the diagonal elements are independent of the order. Theorem 3 ensures that the sweeps from Theorem 2 are defined regardless the order the sweeps are carried out. We used the determinant test to ensure the Within Cross-product matrix and Total Cross-product matrix met the positive definite matrix requirements of Theorem 3.

In stepwise discriminant analysis, we use the sweep operator to control the variable selection process. Jennrich R. I. (1977b:78) describes the creation of two "current status matrices", the within current status matrix $\left(\widetilde{w}_{ij}\right)$ and the total current status matrix $\left(\tilde{t}_{ij}\right)$. The initial values for these matrices are the within sums-of-cross-products matrix $\left(w_{ij}\right)$ and the total sums-of-cross-products matrix $\left(t_{ij}\right)$ respectively.

We now apply these methods to the selection of variables for analysis. In forward stepwise discriminant analysis, we begin with an empty set of variables in our analysis. We use the partial Λ-statistic to determine if x_j should be included in our analysis, shown in Equation 20.

$$V_j = \Lambda\left(x_j * \boldsymbol{x}\right) = \frac{\widetilde{w}_{jj}}{\tilde{t}_{jj}} \tag{20}$$

This corresponds to the F-to-enter statistic, shown in Equation 21, which we use to compute the p-value of the variable under consideration for addition to the model.

$$\mathrm{F_j} = \frac{\mathrm{n-q-p}}{\mathrm{q-1}} * \frac{1-\mathrm{V_j}}{\mathrm{V_j}} \tag{21}$$

Conversely, we use the inverse partial Wilk's Λ-statistic to determine if a variable currently in our model should exit the model. We accomplish this by using Equation 22.

$$V_i = \Lambda(x_i * \boldsymbol{x}') = \frac{\tilde{t}_{ii}}{\widetilde{w}_{ii}} \tag{22}$$

This also corresponds to the F-to-exit statistic, shown in Equation 23.

$$\mathrm{F}_i = \frac{\mathrm{n-q-p+1}}{\mathrm{q-1}} * (V_i - 1) \tag{23}$$

The final consideration in our stepwise discriminant analysis relates to the within group tolerance, or measure of multicollinearity, for the variable x_j not in $\boldsymbol{x}$**.** We measure tolerance using Equation 24 and rearrange Equation 24 to produce the Variance Inflation Factor (VIF) in Equation 25.

$$\mathrm{t_j} = \frac{\widetilde{w}_{jj}}{w_{jj}} \tag{24}$$

$$\mathrm{VIF} = \frac{1}{\mathrm{t_j}} \tag{25}$$

We use Equation 18 through Equation 25 to perform the stepwise discrimination analysis in a three-part process as outlined in Jennrich R. I. (1977b).

1. Remove the variable with the smallest F-to-remove value unless this value is greater than or equal to the F-to-remove threshold (perform inverse sweep on selected variable).
2. If it is not possible to remove a variable, find the variable with the largest F-to-Enter value among all variables whose tolerance is greater than or equal to the tolerance threshold. Enter this variable unless its F-to-enter value is below the F-to-enter threshold (perform sweep on selected variable).
3. If it is not possible to remove or enter a variable the stepping is complete (Jennrich R. I., 1977b:78).

In our analysis, we elected to use a p-value of 0.025, or half the commonly accepted significance level of 0.05, as our threshold measure for entry or exit. This p-value indicates the significance of the change in our Wilk's Λ-statistic from adding or removing the variable (Jennrich R.I., 1977b:77). We selected a p-value of 0.025 because we wanted to ensure the considered variables were extremely significant without overly constricting the variables available for consideration in the stepwise procedures. Additionally, we used a conservative VIF of five as the cutoff for our measure of multicollinearity; again, this is half the frequently accepted threshold (Kutner, Nachtsheim, Neter, & Li, 2005:409).

In backward stepwise discrimination, we begin the stepwise process with a full feature set, or a model that includes all variables as shown in Appendix C. We accomplish this by inverting the within sums-of-cross-products matrix (w_{ij}) and the total sums-of-cross-products matrix (t_{ij}) and use the inverted values as the initial values for the within current status matrix $(\widetilde{w}_{ij})$ and the total current status matrix $(\tilde{t}_{ij})$. Once the within current status matrix $(\widetilde{w}_{ij})$ and the total current status matrix $(\tilde{t}_{ij})$ have been

inverted we simply apply the three-part process mentioned above to build a model using the stepwise discrimination analysis.

Limitations exist with forward and backward stepwise analysis. Kutner, Nachtsheim, Neter, & Li, (2005:368) highlight the fact that forward and backward stepwise analysis methods single out a model as "best" and may become stuck in local optimum solutions. To combat this limitation, we introduced an element of randomness when evaluating variables to include in the model. We adopted the work of Doak (1992), by incorporating the concept of Random Generation Plus Sequential Selection.

Doak (1992) showed that a feature space could be explored using randomly generated subsets and evaluating this subset through Forward Sequential Selection (FSS) and Backward Sequential Selection (BSS). We implemented this model with a slight modification; instead of FSS and BSS, we applied Stepwise Discriminant Analysis. Meaning, we began by randomly selecting the initial size of the empty set from zero to 39, the total number of variables in our analysis. We then introduced randomly selected variables to fill the randomly generated empty set. Finally, we followed the three-part process outlined earlier for Stepwise Discriminant Analysis. Doak (1992:29) found 10 generations sufficient to explore the feature space. We selected a much more conservative 25 generations of the modified RGSS to ensure adequate coverage of the feature space. In theory, by randomly selecting the starting location within the feature space, we reduce the risks of the model finding a single local optimum solution and provide several potential optimal solutions.

Model Selection

After each step in the stepwise discriminant analysis outlined earlier, we evaluated the model using the apparent error rate (APER). This measure of performance evaluates how well the model performs on the training data. We define the APER as the fraction of misclassified observations over the total number of observations in the training set. Figure 6 and Equation 26 illustrate the APER as illustrated in Johnson & Wichern (2007:598-599).

		Predicted Class	
		π_1	π_2
Actual Class	π_1	n_{1C}	$n_{1M} = n_1 - n_{1C}$
	π_2	$n_{2M} = n_2 - n_{2C}$	n_{2C}

Where:
n_{1c} = number of π_1 items correctly classified as π_1 items
n_{1M} = number of π_1 items misclassified as π_2 items
n_{2C} = number of π_2 items correctly classified as π_2 items
n_{2M} = number of π_2 items misclassified

Figure 6. Classification Matrix for the Apparent Error Rate Johnson & Wichern (2007:598)

We then define the APER in Equation 26 as

$$\text{APER} = \frac{\text{n}_{1\text{M}} + n_{2M}}{n_1 + n_2} \tag{26}$$

Using each variable selection method (Forward, Backward, modified RGSS), we recorded data after each step and document the step history. This data consisted of iteration number, smallest p-value to enter, largest p-value to remove, APER, and the list of variables included in the model for that iteration number.

In our analysis, we have no prior data to suggest which variable selection method, if any, provides superior results. We elected to evaluate the top two models from each variable selection method for evaluation. We determined the top two models within each category by first identifying all models whose p-values to enter was larger than the threshold to enter of 0.025 and largest p-value to exit was smaller than threshold to exit of 0.025. Next, we select the two models with the smallest APER overall for validation. We repeat this method for each variable selection method and select the model with the smallest APER. In the event of a tie between models, we incorporate F measure.

F measure serves as an evaluation method used in the field of Information Retrieval and consists of two components. (Manning, Raghavan, & Schutze, 2008:142-144). First, we consider Recall in Equation 27.

$$Recall = R = \frac{n_{1c}}{n_1} \tag{27}$$

where n_{1c} is the number of observations correctly identified as high-risk and n_1 is the total number of observations identified as high-risk. Next, we consider Precision in Equation 28.

Equation 28 Precision

$$Precision = P = \frac{n_{1c}}{n_{1c} + n_{1M}} \tag{28}$$

where n_{1c} is the number of observations correctly identified as high-risk and n_{1M} is the number of observations belonging to the high-risk class but incorrectly identified as nominal risk. We use the weighted harmonic mean of Recall and Precision to produce the F measure in Equation 29. Let

$$F = \frac{1}{\alpha\frac{1}{P} + (1-\alpha)\frac{1}{R}} = \frac{(\beta^2+1)PR}{\beta^2 P + R} \quad where \quad \beta^2 = \frac{1-\alpha}{\alpha}, \tag{29}$$

$\alpha \in [0,1]$, $\beta^2 \in [0, \infty]$, P is the Precision value from Equation 28 and R is the Recall value from Equation 27 (Manning, Raghavan, & Schutze, 2008:144). For further information on the reasoning for the harmonic mean vice the arithmetic mean we direct the readers to (Manning, Raghavan, & Schutze, 2008:144).

We believe the desired risk detection model detects a large proportion of the problems that occur while minimizing the number of false detections. With this in mind, we elected to emphasize precision by choosing a value of $\beta = 0.5$ where values of $\beta < 1$ emphasize precision and values of $\beta > 1$ emphasize Recall. By selecting $\beta = 0.5$, we weight Precision twice as much as Recall. We select the model with the highest F measure to go forward for validation.

Validation

Once we have the models selected, we must evaluate the performance of the models beyond the training data. To accomplish this, we turn to the validation data partitioned prior to the model-building portion of our analysis. As discussed in the *Validation Set* section earlier, we decided to use two validation methods.

First, we validate our selected models against a 20% withhold validation set. We apply the classification function to each observation in our validation set. Finally, we record the performance of the selected model on the validation set using a classification

matrix as displayed in Figure 6. We then calculate APER and Recall as defined in Equation 26 and Equation 27.

Secondly, we then combined the training data and validation data into one dataset and evaluated our model using the holdout procedure process. Lachenbruch's holdout procedure is a four-step process outlined in Johnson & Wichern (2007):

1. Start with the π_1 group of observations. Omit one observation from this group, and develop a classification function based on the remaining $n_1 - 1, n_2$ observations.
   Where:
   $\pi_i = population\ i$
   $n_i = number\ of\ observations\ from\ population\ i$
2. Classify the "holdout" observation using the function constructed in Step 1.
3. Repeat Steps 1 and 2 until all of the π_1 observations are classified. Let $n_{1M}^{(H)}$ be the number of holdout (H) observations misclassified in this group.
4. Repeat Steps 1 through 3 for the π_2 observations. Let $n_{2M}^{(H)}$ be the number of holdout observations misclassified in this group (Johnson & Wichern, 2007:599-600).

Using the holdout procedure method, we calculate the expected actual error rate, $\widehat{E}(AER)$, Equation 30. The expected actual error rate reflects the long-term error rates we would expect over an extended period beyond the data currently available for analysis. Once we evaluated our model using the LOOCV method we looked to expand the definition of high-risk and apply this method to increase utility to the analyst.

$$\hat{E}(AER) = \frac{n_{1M}^{(H)} + n_{2M}^{(H)}}{n_1 + n_2} \tag{30}$$

Multivariate Classification - Alternative Parameterization

We have previously proposed the Multivariate Classification method as an alternative to prior works (Dowling, 2012; Dowling, Miller, & White, 2012; Miller, 2012) in detecting programs expressing high-risk profiles. We now discuss an alternative parameterization of high-risk programs. First, we propose a fundamental change to the definition of high-risk where the EAC must increase over 5% and eliminate the lower boundary of -5%. This more closely aligns the definition of risk in our analysis with that of the Risk Management Guide for DOD Acquisition (OUSD(AT&L), 2006:1), which focuses on the negative consequences of risk. Secondly, we tested the impact of extending the time horizon for identifying high-risk programs from 6-months to 12-months out. This provides the Program Managers more time to react to indicators showing an increase in the risk profile of their programs. These new parameters do not materially change the methodology but simply changes the definitions of the classes and the calculation for change in EAC.

EAC change greater than 5%

Changing the definition of high-risk program class has little impact on our methodology. We accomplished the multivariate classification analysis using the new definition of high-risk programs and nominal risk programs. The methodology does not change due to a change in the labeling of the observations.

Extended time horizon

In this model, we continued the use of the new definition of high-risk programs discussed earlier and attempted to extend the identification horizon to 12-months. This

change resulted in a decrease in the size of the training data. When calculating the 12-month change we evaluate $\Delta\% = \frac{t_{i+12}-t_i}{t_i}$, where t_i is the current observation. This results in a 12-month decrement from each program. Prior to separating a 20% validation set, the 12-month decrement resulted in a database consisting of 816 observations. Given the new database and validation set, we simply analyzed the data using the Multivariate Classification method previously discussed to select a model.

Multinomial Naïve Bayes Classifier

Database

The construction of this database follows the same methods used for the Multivariate Classification method previously mentioned. Meaning, we include the same programs selected for analysis in the Multivariate Classification method in the Multinomial Naïve Bayes Classifier. Instead of using Format 1 data, we now observe the Format 5 data collected from DCARC.

Data Collection

The Multinomial Naïve Bayes Classifier, as previously discussed, constructs a classification model from text in documents available for analysis. To enable this analysis, we first constructed a vocabulary, V, of all words used in the documents, $\mathbb{D}$, of interest. Next, we tokenize, or refine, the vocabulary for use in our analysis. Once we finalized the vocabulary, we observe the count of each word for every month of observation. We further discuss these processes here but many of these operations overlap and accomplished simultaneously.

Vocabulary extraction

We begin vocabulary extraction with documents in many different formats including: Portable Document Files (PDF), Hyper Text Markup Language (HTML), Microsoft Excel® (Microsoft, 2010a), and XML. As previously mentioned, we are unable to address the programs in the XML file format in this analysis. We convert all other Format 5 file formats to Text files (TXT). Using the free statistical software R® (The R Foundation for Statistical Computing, 2011), we create a Comma-Separated Values (CSV) file for each program that combines all monthly observations into one file. Reference Appendix D for an example of the R code used in this analysis. During the execution of this code, we eliminate punctuation and case-fold all words. Case-folding reduces all words to lowercase so that all instances of a particular word can be counted properly (Manning, Raghavan, & Schutze, 2008:28). For example, we count the word *Program* as a separate word from *program* without case-folding. Prior to running the R code, we must remove apostrophes and quotation marks from the TXT files. These characters affect the ability of R to break the text into individual words. We refer the readers to Appendix E for an Excel Visual Basic Application (VBA) code that automates the removal of the previously mentioned special characters.

The program specific CSV file contains columns for every month of observations for the program and rows for each instance of a word in the document with a corresponding count of the word for each column. Figure 7 displays portion of one such CSV file.

sample	AEHF1	AEHF2	AEHF3	AEHF4
	4119	4800	10884	6481
a	51	44	397	221
aassembly	1	0	0	0
ab	1	0	2	1
aborts	1	0	0	0
above	4	1	3	4
ac	1	0	0	0
accelerometer	2	2	2	3
acceptable	1	1	0	1
acceptance	5	1	12	1
access	14	4	5	5
accomplished	1	1	2	1

Figure 7. Program Specific CSV File Screenshot

Following the creation of the 37 program specific CSV files, we create a consolidated CSV file consisting of all programs in our analysis set. We refer the readers to Appendix F for an example of the R code used in this analysis to consolidate all program specific CSV files. In the consolidated CSV file, we include the class labels required for supervised learning. Additionally, we incorporate the *Prgm Name w/ Mo* field; Figure 8 displays a portion of the consolidated CSV file.

	AEHF_1	AEHF_2	AEHF_3	AEHF_4	AEHF_5	AEHF_6
Prgm Name w/ Mo	AEHF	AEHF	AEHF1	AEHF2	AEHF3	AEHF4
a	51	44	397	221	1	493
aa	0	0	1	0	0	1
aaa	0	0	0	0	0	0
aaddjjuusstteedd	0	0	0	0	0	0
aahheeaadd	0	0	0	0	0	0
aanndd	0	0	0	0	0	0
aarrrriivvaall	0	0	0	0	0	0
aassembly	1	0	0	0	0	0
aassttrrootteecchhaassoo	0	0	0	0	0	0
aatt	0	0	0	0	0	0
ab	1	0	2	1	0	3
abandoned	0	0	2	0	0	2

Figure 8. Consolidated Programs CSV Screenshot

Initially, we have 1303 monthly program observations consisting of 37,809 unique words with a total word count of 10,895,076. From Figure 8, we see many of these words make no sense. This occurs due to difficulties arising from the process used by the TXT file format when converting PDF to TXT files (Forman, 2008:263). Additionally, many words are exceedingly rare relative to the total word count. Zipf's Law provides a method to model the distribution of words across documents. Manning, Raghavan, & Schutze (2008) provides the following explination of the Zipf's Law.

> It states that, if t_1 is the most common term in the collection, t_2 is the next most common, and so on, then the collection frequency cf_i of the ith most common term is proportional to $1/i$ (Manning, Raghavan, & Schutze, 2008:82)
>
> $$cf_i \propto \frac{1}{i}$$

Zipf's Law shows that very few words repeat a significant portion of the time. In fact, the frequency drops quickly as i increases, meaning a large proportion of words occur only one to two times in the entire dataset. We find the removal of rare words a common practice when evaluating text data (Forman, 2008:267). While this is a common practice, we find no prescribed threshold to define *rare*. We elected to define *rare* as words with less than five occurrences. This resulted in a removal of 20,003 words or 52.91% of the unique words. Forman (2008) discusses an example where *rare* was defined as less than two occurances and removed nearly half the features, or words (Forman, 2008:267).

In addition to removing rare words, we also remove stop words. Stop words, as previously discussed, are words that are extremely common but provide little information when differentiating between classes. We use a 571-word stop word list created by the

SMART System (Salton, 1971). We find the stop word list available for public use at ftp://ftp.cs.cornell.edu/pub/smart/english.stop. To match the formatting of our analysis we removed apostrophes from the stop word list, which reduced the unique word count on the stop word list to 563 words. We removed 489 words from our dataset related to those on the stop word list.

Next, we remove misspelled words from our analysis set. The Format 5s are professional documents; therefore, the documents should have minimal errors in spelling. Any additional words in the analysis increase the complexity of analysis through the consideration of irrelevant words. We enter the current vocabulary into Word® and use VBA to separate words identified as misspelled by Word® (Microsoft, 2010b). See Appendix G for the VBA code for this operation. By first sorting the words identified as misspelled by frequency, we search through the most commonly misspelled words for words commonly used in EVM analysis. We consider the possibility that high frequencies of misspelled words may represent deliberate usage. Meaning, a word identified as misspelled with a high frequency of usage should remain for consideration due to its accepted use in EVM analysis. For example, Word® identifies *eac* as a misspelled word, due to case folding, when we know this word as an accepted acronym used frequently in EVM analysis. We identify 45 words as exempt from the list of 9,108 words identified as misspelled. Reference Appendix H for a list of words exempted words. After removing rare words, stop words, and misspelled words from our analysis our dataset consisted of 1,303 monthly observations with 8,337 unique words and a total word count of 5,876,740. This initial screening of words represents a 77.89% reduction in

the number of unique words for analysis, but only a 53.94% reduction in total word count.

Based on the performance improvement found in Dowling, Miller, & White, (2012), we began the multinomial Naïve Bayes classifier with the intent to incorporate the model developed from the multinomial Naïve Bayes classifier into the Multivariate Classification model. To ensure the validity of this approach, we identified the 201 monthly CPRs in our dataset used in the Multivariate Classification validation set. Once we identified these observations, we excluded them from our dataset. This ensured that information from the final validation set does not contaminate our model building process thus providing an unfair advantage during validation. After this, we also removed the first two months of observations from each program. We use these two observations in combination with the third month for the standard deviation calculations mentioned in Table 7. Additionally, we discount six months of observations from the end of each program used to calculate the 6-month change and subsequently label the data in the appropriate class.

From the reduced sample, we now have 808 monthly program observations to construct the validation set and training set used in the multinomial Naïve Bayes classifier. Using JMP®, we randomly select 20% of the observations for a validation set. This provides two validation sets and our final LOOCV method. From this multistage validation method we gain insight into the learning behavior of the multinomial Naïve Bayes model as we continue to add more data.

Limitations

In July of 2012, the Office of the Under Secretary of Defense for Acquisition, Technology, and Logistics (OUSD(AT&L)) provided guidance on the Integrated Program Management Report (IPMR). As previously mentioned, the IPMR contains Formats 1 through Format 5. In this guidance, OUSD(AT&L) requires contractors submit Format 5s in a "human readable" format (Department of Defense, 2012c). While all 1303 monthly CPRs contained human readable files, not all files contained *searchable text* files. In other words, a human reader is capable of observing the file, reading, and interpreting the text, but when we attempt to convert the file to a TXT file, the computer is unable to recognize the text in a meaningful way. This problem affects 34 monthly CPR files in our dataset. In cases where the words of a document provide no clear evidence for one class or another, we use the prior probability of a document occurring in class *c* to classify the document (Manning, Raghavan, & Schutze, 2008:239). We also use this method for the 10 data gaps found in the EVM dataset; in the multivariate classification, we addressed these gaps using linear interpolation.

We see from Table 3, in Chapter II, analysis of Format 1 data follows specific formulas and structured analysis. In Format 5 data, there is very little consistency in form or function of the reports. Some programs meet the intent of the Format 5, as described in Table 2, by providing PDF documents consisting of charts and detailed variance analysis. Others simply have an Excel sheet that provides a short explanation of variances experienced by the program. We compiled all sections labeled Format 5 in each program, instead of attempting to comb through all 1303 documents to identify directly comparable

sections of the Format 5 across programs. This method may have inadvertently introduced noise features, or words that may increase the classification error for new observations (Manning, Raghavan, & Schutze, 2008:251). Manning, Raghavan, & Schutze (2008) argue the multinomial Naïve Bayes classifier is robust against these noise features minimizing the impact of this limitation on our analysis (Manning, Raghavan, & Schutze, 2008:249).

Multinomial Naïve Bayes Classification Model Building

In Chapter II, we outlined the application of the Naïve Bayes Classifier. Here we discus two methods we applied to improve the performance of the Naïve Bayes Classifier prior to validation. First, we provide our approach to add-α smoothing, and then transition to specify our feature selection methodology. We then discuss the analysis of the development data, and end our model building discussion by describing our model selection process and validation of the selected model.

Add-α smoothing

Earlier, we discussed the generalization of the Laplace Smoothing to add-α smoothing in Equation 10. We were unable to find conclusive support for a single value when applying add-α smoothing. To ensure a thorough analysis, we systematically explore different levels of α for inclusion in the final model. We develop a baseline of performance by applying the Laplace smoothing, as defined in Equation 9. Next, we explored the effect the value of α on the Naïve Bayes classifier by testing a wide spread of values. We use Equation 31 to calculate our value for α and later provide further detail

concerning the application of these values in our discussion on analysis of the development data. Let

$$\alpha = \left(\frac{1}{4}\right)^{i-1}, \text{ where } i = 1, \dots, 8 \tag{31}$$

In addition to add-α smoothing, we also consider the impact the number of words, or features, included in our analysis. When evaluating the impact of words for consideration in the model, we use feature selection. Feature selection seeks to accomplish two goals. First, feature selection seeks to improve the efficiency of the Naïve Bayes classifier by reducing the number of words in the vocabulary. Secondly, feature selection improves accuracy by reducing the number of noise features in the vocabulary (Manning, Raghavan, & Schutze, 2008:251).

Feature Selection

Within the field of machine learning, there are many methods available for feature selection (Liu & Motoda, 2008). We selected mutual information (MI), a common feature selection method, for use in this analysis (Manning, Raghavan, & Schutze, 2008:252-255). "MI measures how much information the presence/absence of a word contributes to making the correct classification decision on *c*" (Manning, Raghavan, & Schutze, 2008:252). We calculate the MI of a word *t* represented by the random variable U in some class C by evaluating Equation 32.

$$I(U;C) = \frac{N_{11}}{N} log_2 \frac{NN_{11}}{N_{1.}N_{.1}} + \frac{N_{01}}{N} \log_2 \frac{NN_{01}}{N_{0.}N_{.1}} + \frac{N_{10}}{N} \log_2 \frac{NN_{10}}{N_{1.}N_{.0}} + \frac{N_{00}}{N} \log_2 \frac{NN_{00}}{N_{0.}N_{.0}} \quad (32)$$

where the *N*s are the counts of documents that contain the values $e_t = 1$ (for the documents containing word t), $e_t = 0$ (for the documents not containing word t), $e_c = 1$ (the document is in class *c*), and $e_c = 0$ (the document is not in *c*) (Manning, Raghavan, & Schutze, 2008:252). To clarify further, we have provided an example below.

	$e_c = e_{high-risk} = 1$	$e_c = e_{high-risk} = 0$
$e_t = e_{abandoned} = 1$	$N_{11} = 7$	$N_{10} = 4$
$e_t = e_{abandoned} = 0$	$N_{01} = 199$	$N_{00} = 437$

$$I(U;C) = \frac{7}{647} \log_2 \frac{647 * 7}{(7+4)(7+199)} + \frac{199}{647} \log_2 \frac{647 * 199}{(199+437)(7+199)}$$
$$+ \frac{4}{647} \log_2 \frac{647 * 4}{(7+4)(4+437)} + \frac{437}{647} \log_2 \frac{647 * 437}{(199+437)(4+437)}$$
$$\approx 0.005304$$

We apply this calculation to each of the words in our vocabulary and record the resulting MI value. Following the MI calculations, we must decide how many words to include in the analysis. Manning, Raghavan, & Schutze (2008:251) detail a feature selection algorithm that returns *k* words. Due to the varied application of MI thresholds, we find no generally accepted threshold for MI or a recommended number of *k* words to include in the analysis. In the absence of clear guidance, we explored a range of possible minimum MI thresholds.

An example provided by Manning, Raghavan, & Schutze (2008:253:254) defines high MI values ranging from as high as 0.19 to as low as 0.0004. We believe testing values between 0 and 0.01 by 0.001 increments provides the appropriate level of analysis for our applications. Each increment increase in the MI threshold increases the required information a word must contribute represented by MI for a specific word. This restriction reduces the number of words available in our analysis vocabulary. This reduction in vocabulary leads to improved efficiency of the Naïve Bayes classifier and reduces the noise features, the two goals of feature selection previously discussed.

We propose an additional constraint on the MI feature selection method. When a rare word, such as forge, contains no relevance to a specific class, for example Nominal Risk, but by chance all instances of the word from our training set fall in the Nominal Risk category, we may produce a classifier that incorrectly assign documents to a class. Manning et al. defined this accidental occurrence as overfitting (Manning, Raghavan, & Schutze, 2008:251). In our example, this overfitting results in a maximum MI; thus we combat the problem of overfitting by requiring any word observed in only one class occur in more than 5% of the total number of documents in our dataset. If the word does not meet the 5% criteria, we do not consider the word in our analysis. This additional criterion reduces, but does not eliminate, the risk of overfitting by reducing the chances the word falls into a specific class accidentally.

Model Development

Once we determined the test values for add-α smoothing and MI thresholds, we turned our attention to analysis of the training data. We accomplished this analysis by

developing potential models for each value of α given a specific value for MI threshold using Equation 33.

$$c_{map} = \underset{c \in \mathbb{C}}{argmax} \left[\log \frac{N_c}{N} + \sum_{1 \le k \le n_d} \log \frac{T_{ct} + \alpha}{(\sum_{t' \in V} T_{ct'}) + \alpha B'} \right] \tag{33}$$

where N is the total number of documents, N_c is the total number of documents belonging to class *c*, T_{ct} is the number of times the word *t* appears in the training document from class *c*, α is our smoothing value from Equation 31, and $B = |V|$ is the cardinality, or number of words after applying the MI reduced vocabulary, of the training data. We record the predicted class using the classification matrix from Figure 6. The resulting classification matrix serves as inputs to a table with headings listed in Table 11. Upon completion of the model development phase, we produce 88 different models, one for each value of α given a specific threshold for MI.

Table 11. Model Performance Headings

MI Threshold	Error Rate = Errors/N	High-Risk given High-Risk = n_{1C}
Word Count = |V|	Nominal Risk | Nominal Risk = n_{2C}	Recall (see Equation 27)
α value	Nominal Risk | High-Risk = n_{1M}	Precision (see Equation 28)
Errors = $n_{1M} + n_{2M}$	High-Risk | Nominal Risk = n_{2M}	F measure (see Equation 29

Model Selection

From Table 11, we use Recall, Precision, and F measure to evaluate each model under consideration. We previously discussed the application of Recall, Precision, and F measure in our discussion on the multivariate classification model selection. To reiterate,

the Recall measures the fraction of observations classified as High-Risk that belong to the High-Risk population. Precision measures the number of observations belonging to High-Risk population classified as High-Risk by the classification model. F measure, our final evaluation criteria combines Recall and Precision using a weighted harmonic mean. We evaluate the F measure with a $\beta = 0.5$. Again, this emphasizes Precision in support of correctly identifying a large proportion of the High-risk programs while minimizing the false detections. We select the model with the highest F measure to go forward for validation.

Validation

Once we have a selected model, we perform the same validation method outlined for the multivariate classification model. We accomplish this by first applying our selected model to each observation and record the predicted class in a classification matrix as in Figure 6. We then calculate Recall as defined in Equation 27. Secondly, we combined the training data and validation data into one dataset and evaluated our model using the holdout procedure. We then follow the holdout procedure process detailed earlier. We record the results of the holdout procedure in a classification matrix and use these results to calculate $\widehat{E}(AER)$ as defined in Equation 30 and Recall. Again, the expected actual error rate reflects the long-term error rates we would expect over an extended period beyond the data currently available for analysis. The final step in our validation method, included validation against the multivariate classification withhold. This serves two purposes. First it provides the multinomial Naïve Bayes input to the

hybrid model discussed shortly. Secondly, we see the performance of the model using additional data and a separate validation set.

Multinomial Naïve Bayes Classifier – Alternative Parameterization

The alternative parameterization of the multinomial Naïve Bayes classifier mirrors those proposed for the multivariate classification model. First, we sought to redefine the high-risk program class allowing for a more accurate reflection of the DOD accepted definition of risk. Secondly, we tested the impact of extending the risk identification period from 6-months to 12-months.

EAC change greater than 5%

Our analysis methods do not change due to a change in the definition of the high-risk program class. We applied the same methodology already discussed for the multinomial Naïve Bayes classifier using the new labels. Again, this produces risk categories that more closely align with the DOD definition of risk.

Extended time horizon

In this model, we continued the use of the new definition of high-risk programs discussed earlier and attempted to extend the identification horizon to 12-months. Prior to separating a 20% validation set, the 12-month decrement resulted in a database consisting of 816 observations. Given the new database and validation set, we analyzed the data using the multinomial Naïve Bayes classifier method previously discussed to select a model for validation. We execute the same validation methodology provided earlier for the multinomial Naïve Bayes classifier.

Hybrid Multivariate Classification and Multinomial Naïve Bayes Classifier

Dowling, Miller, & White, 2012 introduced the idea of combining data from the Format 1 and Format 5 using Statistical Process Control Methods. Dowling et al. accomplished a unified model by using a weighted average of the model outputs of Dowling (2012) and Miller (2012). In Figure 2, we see this unified model provided better outputs than either model on its own; specifically, we saw an 19.96% and a 24.61% improvement in the probability of correctly identifying high-risk programs respectively.

We propose an alternative hybrid model using our multivariate classification and multinomial Naïve Bayes classifier. Initially, we continue the use of the 6-month detection timeframe as those used by Dowling et al. (2012) for comparability. However, we later we discuss alternative parameterization for this method as well.

Our hybrid model begins by applying the validated multinomial Naïve Bayes classifier from our earlier analysis. For each observation, we collect the predicted class as assigned by the Naïve Bayes classifier and introduce a new variable, *NB_Pred_Class*, to the multivariate classification variable list. We characterize this variable as a categorical variable with a value of 1 if the multinomial Naïve Bayes classifier predicted the observation a high-risk program and 0 otherwise. We match the monthly CPR in the multinomial Naïve Bayes classifier with the appropriate monthly CPR in the multivariate classification model.

With the new categorical variable included in the multivariate analysis, we execute the multivariate classification as discussed in the multivariate classification section earlier. This includes performing the forward stepwise discriminant analysis,

backward stepwise discriminant analysis, and modified RGSS. We select the best performing model as outlined in the multivariate classification section.

In the vocabulary extraction section of our discussion on the multinomial Naïve Bayes Classifier, we explained a partitioning of the multivariate classification validation set from the rest of the data considered. This data provides an opportunity to validate our hybrid approach and measure its performance. We begin by applying the multinomial Naïve Bayes classifier to the validation set. Next, we record the predicted classes and introduce the new categorical variable to the multivariate classification validation set. We validate the selected best performing hybrid classification model against both the 20% withhold and Lachenbruch's holdout procedure.

Alternate Hybrid Model Parameterization

As with the previous alternate model parameterization, we redefine the high-risk class and now evaluate the 6-month model only looking for programs expected to experience cost growth of greater than five percent. Additionally, we extend the timeframe of our high-risk program detection from the original 6-months to 12-months. Again, we use the new definition of high-risk programs as those programs expected to experience cost growth of greater than five-percent. Following each of the new parameterizations, we execute the analysis in the same way outlined in the hybrid multivariate classification and multinomial Naïve Bayes classifier section.

Summary

In this chapter, we provided a detailed description of the analysis conducted for this study. We discussed the four distinct components for our analysis. First, we began

detailing the analysis of EVM data using Multivariate Classification techniques to identify high-risk acquisition programs. Secondly, we introduced the multinomial Naïve Bayes classification technique on the Format 5 data to identify high-risk programs. Next, we detail the hybrid model consisting of the Multivariate Classification technique and the multinomial Naïve Bayes classifier to produce a new risk detection method. Lastly, alternative parameterization for each component of the analysis provided a realignment of our definitions of risk to those accepted by the DOD and provided an improved lead-time to administer mitigation plans for programs identified as high risk. In the next chapter we show the results of our analysis.

IV. Analysis and Results

In previous chapters, we discussed the methods applied to our dataset, the literature that supports these methods, and outlined the research questions we sought to answer. Here we provide empirical evidence showing the viability of these alternative risk detection methods. We accomplish this in three parts. First, we present models for identifying programs at risk of a 6-month cumulative change in EAC of greater than 5% in magnitude. Next, we discuss the 6-month models seeking to identify programs at risk of a cumulative increase in the EAC of 5%. We then provide our results from extending the identification timeframe from 6-months to 12-month of programs at risk of a cumulative increase in the EAC of greater than 5%. Finally, we conclude this chapter presenting the single best performing model for each definition of high-risk programs.

6-month Risk Models (Cumulative Change of Greater Than 5% in Magnitude)

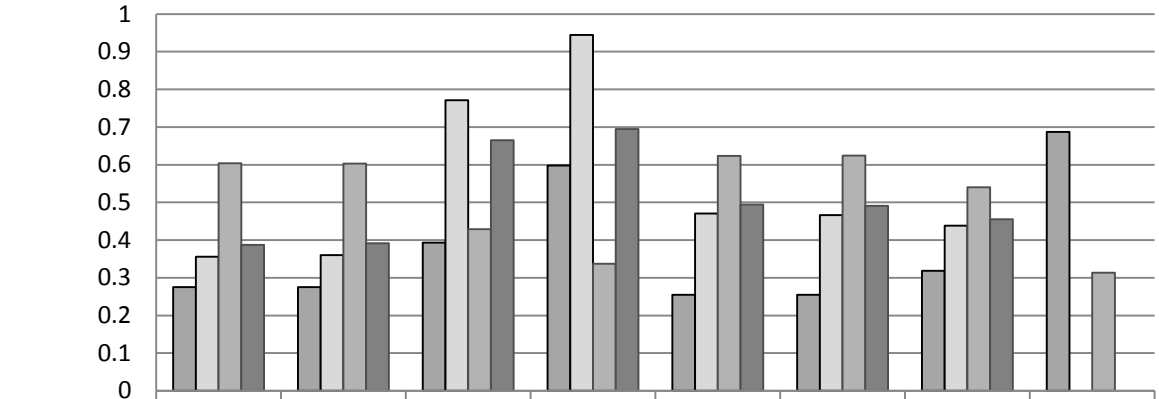
In this section, we present our results specific to the identification of programs classified at high risk of experiencing a cumulative change in the EAC of greater than 5% in magnitude six months from the current observation. We begin by outlining the results from our multivariate classification model. Next, we transition to the results associated with the multinomial Naïve Bayes text classifier. We then display the results provided by the hybrid classification model. Finally, we provide a summary of the validated models for each method.

Multivariate Classification Results

In the Multivariate Classification Model Building section of Chapter III, we proposed three stepwise variable selection methods for developing potential models. We begin this section by detailing the results of the forward stepwise discriminant analysis, backward stepwise discriminant analysis, and conclude with the results from the modified RGSS. In Figure 9, we provide a side-by-side comparison of the top two potential models from each of these selection methods. We seek lower APER values but higher Precision, Recall, and F measure values. For comparison purposes, we included the training set results from Dowling (2012) converted to match our detection of high-risk programs, or the cumulative change over six months.

We caution the readers, Dowling (2012) optimized his model to identify programs at risk of a one-month change in the EAC greater than 5% in magnitude within six months. However, we optimized our models to detect the cumulative change in EAC in exactly six months from the current observation. This implies the results may not allow a direct comparison but still provides the closest proxy model available for comparison.

Additionally, we include an overly simplistic model in which we classify all monthly CPRs high-risk. By doing so, we provide a baseline comparison which we can use to determine if any model can improve on this untrained classification. By default, this model will score a perfect one for Precision due to the simplistic classification rule; therefore, we believe the Precision and F measure for the All High-Risk model provides no useful comparison but we call the reader's attention to the APER and Recall for useful comparisons.



	Forward Stepwise1	Forward Stepwise2	Backward Stepwise1	Backward Stepwise2	Modified RGSS1	Modified RGSS2	Dowling (2012)	All High-Risk
APER	0.2748	0.2748	0.3936	0.5978	0.2550	0.2550	0.3188	0.6869
Precision	0.3557	0.3597	0.7708	0.9447	0.4704	0.4664	0.4380	
Recall	0.6040	0.6026	0.4286	0.3376	0.6230	0.6243	0.5405	0.3131
F measure	0.3876	0.3912	0.6646	0.6948	0.4946	0.4913	0.4552	

Figure 9. Multivariate Classification Model Comparison

As Figure 9 shows, the performance of the modified RGSS models strictly dominate all other methods of model selection for both APER and Recall measures. The forward selection method produces identical APER measures for the top two models but differs slightly in the quality of the Recall measure. We see the same effect in the modified RGSS models. The differences between these models traces their roots to the variables selected for each model. We show in Table 12, the performance of each model and the variables that comprise each one. It becomes clear, predictive quality of each variable influences performance and not the total number of variables included in the model. For example, simply comparing Backward Stepwise1 and Backward Stepwise2, which differ by one variable, represents a 51.88% increase in APER from Backward Stepwise 1 to Backward Stepwise2.

Table 12. Multivariate Classification Model Output

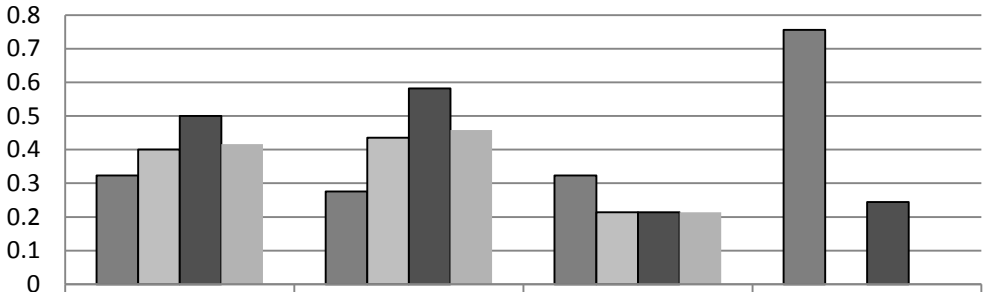
	Forward Stepwise1	Forward Stepwise2	Backward Stepwise1	Backward Stepwise2	Modified RGSS1	Modified RGSS2
Generation					3	3
Iterations	10	11	25	26	64	67
P-Value to Enter	0.009576634	0.015903805	0.00704813	0.059177116	0.011115686	0.022325528
P-Value to Remove	0.002500703	0.009576634	0.01115027	0.014014678	0.014760797	0.01807075
APER	0.274752475	0.274752475	0.393564356	0.597772277	0.254950495	0.254950495
Precision	0.355731225	0.359683794	0.770750988	0.944664032	0.470355731	0.466403162
Recall	0.604026846	0.602649007	0.428571429	0.337570621	0.623036649	0.624338624
F measure	0.387596899	0.391229579	0.664621677	0.694767442	0.494596841	0.491257286
Variable count	8	9	15	16	12	13
Variables	CV%	CV%	CPI	CPI	% Complete	% Complete
	% Difference Between ML and B	% Difference Between ML and B	TSPI	TSPI	CV%	CV%
	CPI 1 Month Change	CPI 1 Month Change	CV%	CV%	% Difference Between W and B	% Difference Between ML and B
	TSPI 2 Month Change	TSPI 2 Month Change	% Difference Between ML and B	% Difference Between ML and B	StDev CPI	StDev CPI
	Joint	Joint	StDev CPI	StDev CPI	StDev SPI	StDev SPI
	Comm.	Comm.	CV% StDev	TCPI StDev	CV% StDev	CV% StDev
	Radar	Facility	CPI 1 Month Change	CV% StDev	SPI 1 Month Change	CPI 1 Month Change
	Small	Radar	SCI 1 Month Change	CPI 1 Month Change	TSPI 2 Month Change	SPI 1 Month Change
		Small	CPI 2 Month Change	SCI 1 Month Change	Comm.	TSPI 2 Month Change
			CV% 2 Month Change	CPI 2 Month Change	Facility	Comm.
			Comm.	CV% 2 Month Change	Radar	Facility
			Facility	Comm.	Small	Radar
			Missile	Facility		Small
			Radar	Missile		
			Small	Radar		
				Small		

In our variable selection discussion in the multivariate classification model building section of Chapter III, we identified limitations associated with both the forward and backward stepwise variable selection method. We see from Figure 9 and Table 12 the modified RGSS' ability to explore more of the feature space allows the identification of higher performing models. As evident in Table 13, modified RGSS proceeds through 25 generations, each generation terminating based on the p-value convergence criterion specified in Chapter III.

Table 13. Multivariate Variable Selection Method Breakdown

Model Type	Generations	Steps	Average Steps per generation
Forward Discriminant Analysis	1	12	12
Backward Discriminant Analysis	1	26	26
Modified RGSS	25	441	17.64

After identifying Modified RGSS1 as our best performing model, we executed the two validation methods discussed in our multivariate classification validation section in Chapter III. We take this opportunity to reiterate the important distinction between the APER and the $\hat{E}(AER)$. The APER shows the performance of the model on data withheld prior to model building. As previously discussed, this withheld data forces us to build a model on data that does not include all available data, thus producing a model that does not represent our entire dataset. We overcome this limitation by executing the Lachenbruch's holdout procedure, or LOOCV, we outlined in Chapter III. This method provides a more representative model of the entire dataset and produces a nearly unbiased estimate of the long-term error rate. As shown in Figure 10, the APER performance of the modified RGSS1 (withhold validation) marginally outperforms the Dowling (2012) proxy model. Additionally, we see modified RGSS1 (LOOCV) significantly outperforms when measured by Recall representing a 171% improvement in the model's ability to identify correctly, programs belonging to the high-risk class when compared to the proxy model.



	Modified RGSS1 (Withhold Validation)	Modified RGSS1 (LOOCV)	Dowling (2012)	All High-Risk
APER	0.3234	0.2758	0.3235	0.7556
Precision	0.4000	0.4353	0.2143	
Recall	0.5000	0.5823	0.2143	0.2444
F measure	0.4167	0.4585	0.2143	

Figure 10. Multivariate Classification Validation Performance

Multinomial Naïve Bayes Classifier Results

In Chapter II and Chapter III, we described the application of the multinomial Naïve Bayes classifier to the Format 5 data from the monthly CPRs. In this section, we detail our results from the application of the aforementioned methods. We begin by outlining the trends identified in our add-α smoothing and MI thresholds. Next, we provide results of the top five models produced by the multinomial Naïve Bayes classifier prior to the partial validation. We conclude this section by providing the validated results from our best performing model using both the partial and full validation datasets.

In Chapter III, we discussed how add-α smoothing and MI thresholds potentially influence the performance of the Naïve Bayes classifier. As Figure 11 shows, we see lower values of α produce, on average, lower error rates in our training set. Additionally, Figure 12 indicates the MI thresholds influence the models in a much more substantial way. Figure 13 shows a sharp decrease in the number of words considered in our models.

We leverage these two performance-improving methods and evaluate all combinations discussed in Chapter III. Once we developed all potential models, we evaluate each model's F measure. Figure 14 provides the top five performing models arranged by F measure. We select Model 65 for validation due to its performance, as measured by its F measure, compared to all other models (see Chapter III, page 54 for discussion on multinomial Naïve Bayes text classifier model selection discussion).

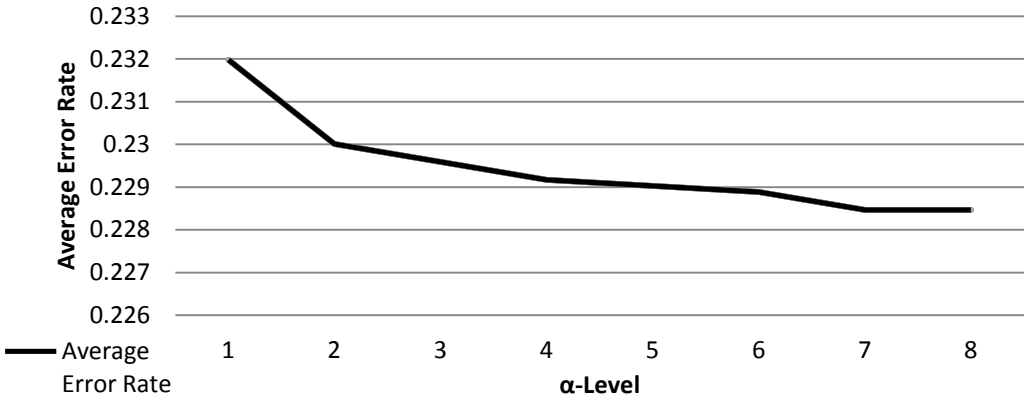


Figure 11. Average Error Rate vs. Add-α Smoothing level for Naïve Bayes text classification, 6-month cumulative change in EAC greater than 5% in magnitude (The x-axis shown as qualitative scaling evaluated at $\alpha = \left(\frac{1}{4}^{i-1}\right)$, $i = 1, \ldots, 8$

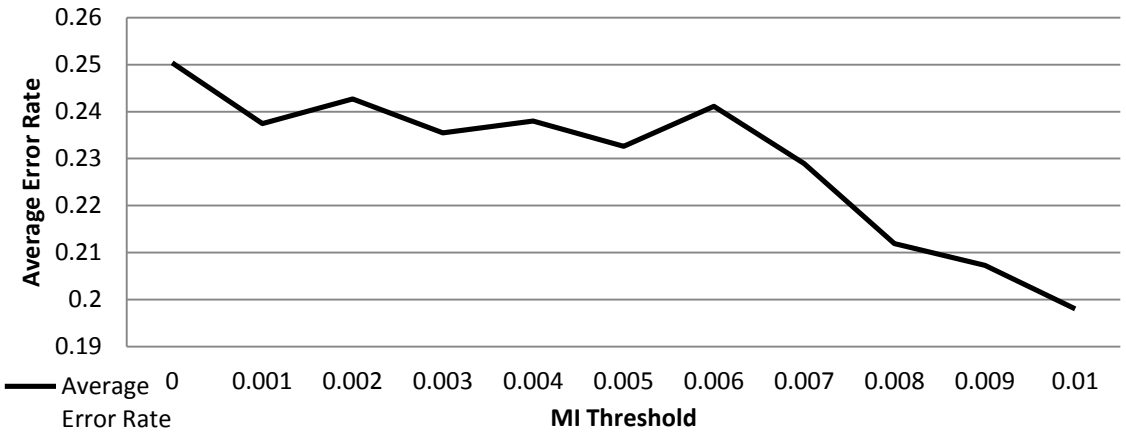
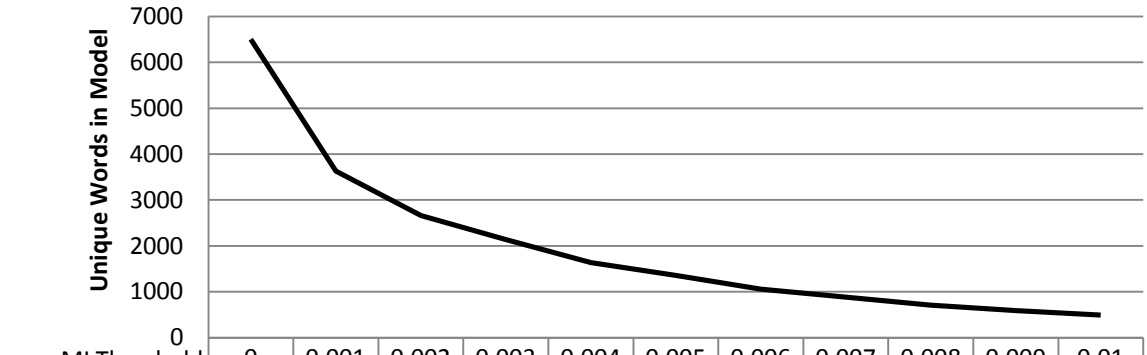
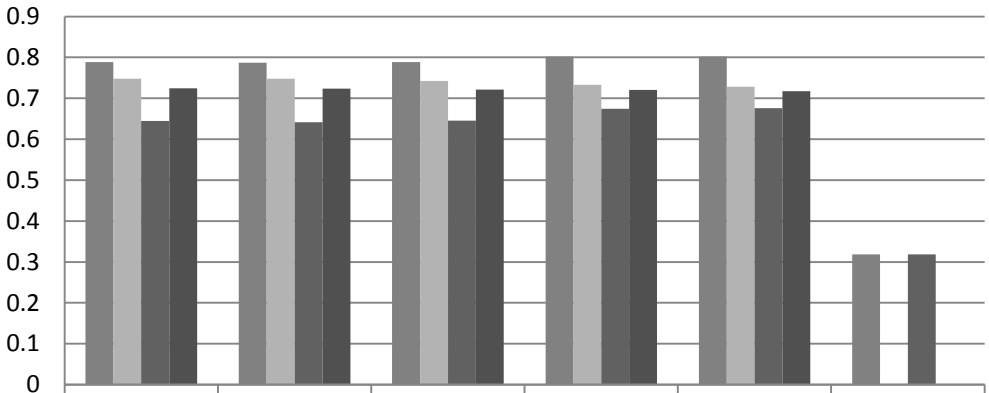


Figure 12. Average Error Rate vs. MI Threshold for Naïve Bayes text classification, 6-month cumulative change in EAC greater than 5% in magnitude



MI Threshold	0	0.001	0.002	0.003	0.004	0.005	0.006	0.007	0.008	0.009	0.01
Word Count	6505	3629	2664	2140	1639	1359	1058	882	708	589	495

Figure 13. 6-Month Model Vocabulary vs. MI Threshold represents decreasing word count as MI increases

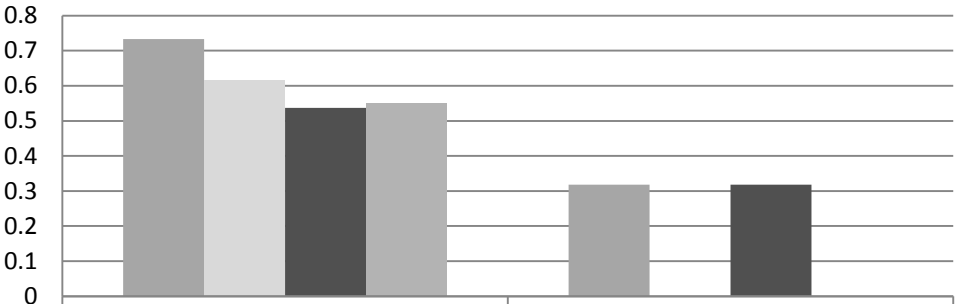


	Model 65	Model 66	Model 67	Model 81	Model 83	All High Risk
Accuracy	0.7883	0.7867	0.7883	0.8022	0.8022	0.3184
Precision	0.7476	0.7476	0.7427	0.7330	0.7282	
Recall	0.6444	0.6417	0.6456	0.6741	0.6757	0.3184
F Measure	0.7244	0.7237	0.7210	0.7204	0.7170	

Figure 14. Multinomial Naive Bayes Text Classifier Model Comparison

In Chapter III, we described the Naïve Bayes text classification data as 80% of the data available for the multivariate classification method. Initially, our training set consisted of 80% of the 80% available and our validation withhold consisted of the 20 % remaining. Here, we refer to this validation as partial validation. Shortly, we discuss the validation of our model against the 20% withhold from the multivariate classification

method; we describe this as full validation. In Figure 15, we show the performance of our model against the partial validation results. We have also provided the simple classification model previously discussed which classifies all monthly CPR observations as high-risk. We see the multinomial Naïve Bayes provides a 69.67% improvement in correctly identifying high-risk programs and a 130% improvement in overall accuracy.

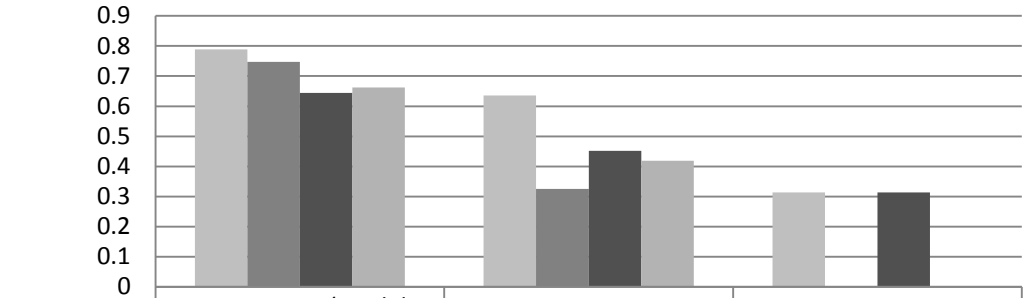


	Model 65 (Partial Validation)	All High-Risk
Accuracy	0.7329	0.3184
Precision	0.6170	
Recall	0.5370	0.3184
F measure	0.5513	

Figure 15. Naive Bayes Partial Validation

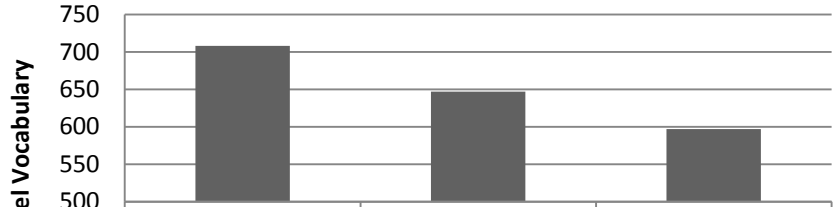
Once we accomplished the partial validation, we found it necessary to validate the model against the full validation set. This provides us several advantages such as the ability to compare results from Miller (2012) with a more compatible scale and provides an opportunity to understand the learning behavior as we add data to the model. We see from Figure 16, the multinomial Naïve Bayes classifier improves on all measures provided. Most significantly, we improve the ability to identify correctly programs belonging to the high-risk class by 43% and improve overall accuracy by 24% over those found in the Miller (2012) proxy model. In Figure 17, we see a downward trend of unique words meeting the MI threshold as we include additional data in the model. This implies

as additional data becomes available, the multinomial Naïve Bayes classifier learns to differentiate further between classes with fewer words meeting the MI threshold.



	Naïve Bayes (Model 65)	Miller (2012)	All High-Risk
Accuracy	0.7883	0.6351	0.3131
Precision	0.7476	0.3248	
Recall	0.6444	0.4518	0.3131
F Measure	0.6627	0.4190	

Figure 16. Text Analysis Full Training Set Model Comparison

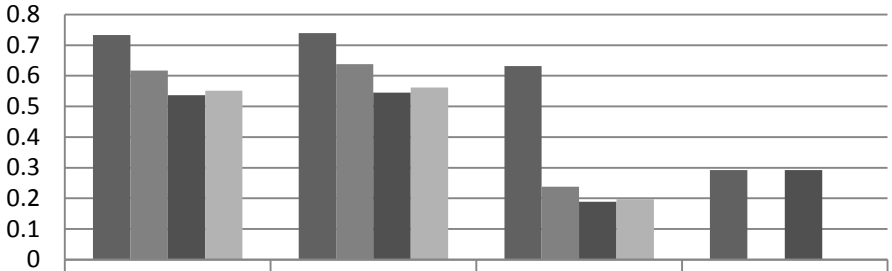


	Partial Validation	Full Validation	LOOCV
Model Vocabulary	708	647	597

Figure 17. Naive Bayes Vocabulary Trends

We again caution the readers by saying Miller (2012) sought to optimize the detection of a one-month change in the EAC greater than 5% in magnitude within six months, much like Dowling (2012). Our detection method seeks to identify these high-risk programs exactly six months from the current observation. However, the numbers associated with Miller (2012) reflect the proxy model adapted from Miller and applies our definition of a successful detection.

We see from Figures 15 and 16, the Naïve Bayes text classifier performs well when validated using the partial validation set and shows improvement when scaled up to the full training dataset. In Figure 18, we provide the results from our full validation set as well as the LOOCV method. It is clear the Naïve Bayes classifier provides a significant advantage over the proxy text classification method and baseline measure. When we compare the results from Figure 18, we see the Precision and Recall measures show a tendency toward stabilizing at these levels. The validated results show a strong performance when compared with the Miller (2012) proxy and All High-Risk models. Specifically, when we compare the ability of Naïve Bayes LOOCV classifier to identify correctly a high-risk program, we see an 87% improvement over the simple untrained classifier and a 189% improvement over the Miller (2012) proxy model.



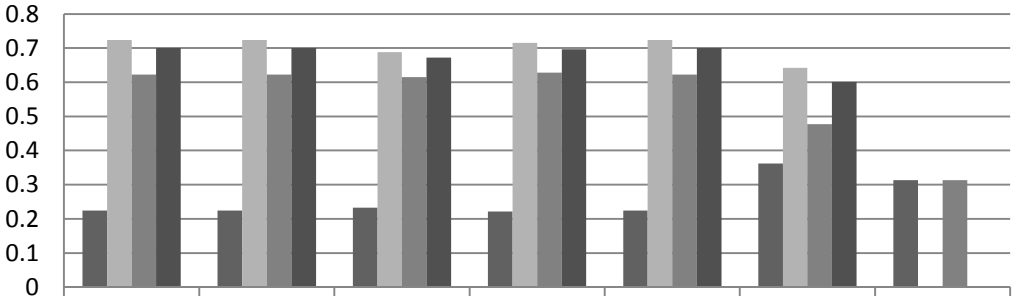
	Naïve Bayes (Model 65 - Withhold Validation)	Naïve Bayes (Model 65- LOOCV)	Miller (2012)	All High-Risk
Accuracy	0.7329	0.7391	0.6324	0.2919
Precision	0.617	0.6383	0.2381	
Recall	0.537	0.5454	0.1887	0.2919
F Measure	0.5513	0.5618	0.1969	

Figure 18. Text Analysis Full Validation Set

Hybrid Multivariate and Naïve Bayes Text Classification Model

In this section, we relay the results from our hybrid model. We accomplish this first by detailing our best performing potential models using the forward stepwise discriminant analysis, backward stepwise discriminant analysis, and modified RGSS. Next, we show the performance of the selected model against the withheld validation data and the LOOCV method. We conclude this section by displaying a summary of the best performing model from the multivariate classification, multinomial Naïve Bayes classifier and our hybrid classification model.

We begin with Figure 19, which shows the top two performing models for each model selection method with one exception. The backward stepwise discriminant method found one significant model prior to meeting the stopping criteria laid out on page 41. Additionally, we include a proxy to the weighted average model produced by Dowling et al. (2012) for comparison purposes and the simple untrained classifier.



	Forward Stepwise1	Forward Stepwise2	Backward Stepwise1	Modified RGSS1	Modified RGSS2	Modified Dowling et al. (2012)	All High-Risk
APER	0.2240	0.2240	0.2327	0.2215	0.2240	0.3624	0.3131
Precision	0.7233	0.7233	0.6877	0.7154	0.7233	0.6423	
Recall	0.6224	0.6224	0.6148	0.6285	0.6224	0.4770	0.3131
F measure	0.7006	0.7006	0.6718	0.6962	0.7006	0.6007	

Figure 19. Hybrid Classifier Model Comparison

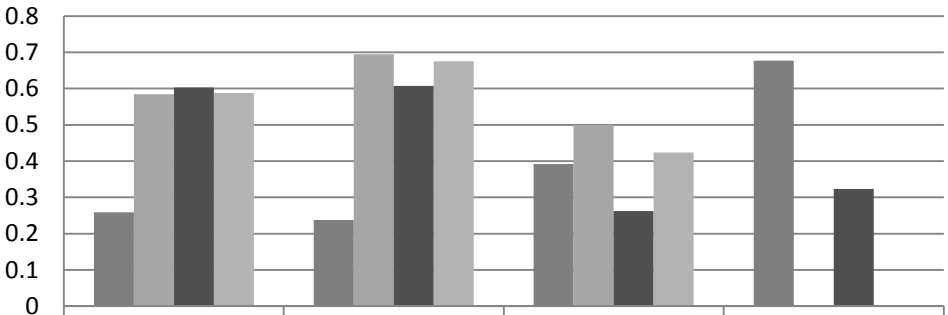
We see from Figure 19, these models tend to perform in very similar fashion. For example, Forward Stepwise1, Forward Stepwise2, and Modified RGSS2 mirror each other. Based on our hybrid model selection criteria (see page 58), we chose modified RGSS1 due to its outperformance of all other models when evaluated on APER. In Table 14, we provide the variable composition for each of the models from Figure 19. A common theme appears when we consider the repetition of variables between models. In Chapter V, we provide further insight to these patterns and our interpretations of them.

Table 14. Hybrid Classifier Model Output

	Forward Stepwise1	Forward Stepwise2	Backward Stepwise1	Modified RGSS1	Modified RGSS2
Generation				11	1
Iterations	1	2	27	140	4
P-Value to Enter	3.56761E-05	0.001311302	0.035161701	0.000982009	3.56761E-05
P-Value to Remove	0	3.56761E-05	0.014053896	0.001988425	0
APER	0.224009901	0.224009901	0.232673267	0.221534653	0.224009901
Precision	0.723320158	0.723320158	0.687747036	0.71541502	0.723320158
Recall	0.62244898	0.62244898	0.614840989	0.628472222	0.62244898
F measure	0.700612557	0.700612557	0.671814672	0.696153846	0.700612557
Variables Count	1	2	14	3	1
Variables	NB_Pred_Class	Small	CPI	TSPI 2 Month Change	NB_Pred_Class
		NB_Pred_Class	TSPI	Small	
			CV%	NB_Pred_Class	
			% Difference Between ML and B		
			StDev CPI		
			CV% StDev		
			CPI 2 Month Change		
			CV% 2 Month Change		
			AF		
			Army		
			Joint		
			Helicopter		
			Small		
			NB_Pred_Class		

We see no difference in performance between Forward Stepwise1, Forward Stepwise2, or Modified RGSS2. From Table 14, we see that Forward Stepwise1 and Modified RGSS2 results in the same one variable model. Additionally, we see Forward Stepwise2 results in a model with two variables but no difference in performance. In Forward Stepwise2, the variable *small* proves statistically significant in discriminating between the two classes, evidenced by a p-value of 0.000036. This significance in discriminating between classes does not provide any additional classifying information beyond that contained in the variable *NB_Pred_Class*. This results in a less parsimonious model than the one variable models seen in Forward Stepwise1 and Modified RGSS2 with identical performance.

Next, we consider the validation results using both the withheld validation data and the LOOCV methods. In Figure 20, we see the results of the hybrid classification provide superior performance to both the modified weighted model produced by Dowling et al. (2012) and the simple untrained classification model. Specifically, by using the LOOCV Hybrid Classification Model we see a 62% reduction in the APER over the All High-Risk classifier and 39% reduction in the APER from that found in the Modified Weighted Model adapted from Dowling et al. (2012).



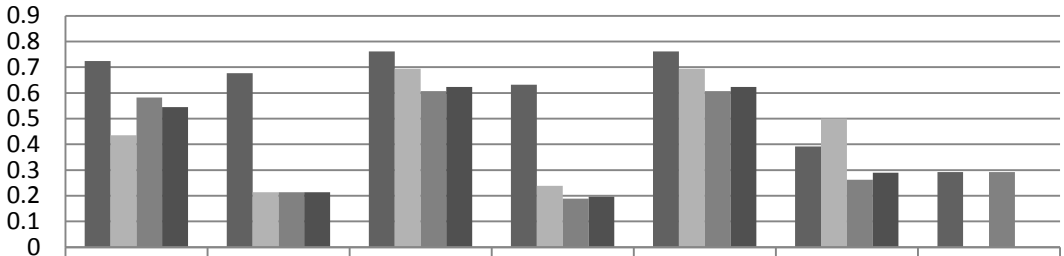
	Hybrid Classification Model (Withhold validation)	Hybrid Classification Model (LOOCV)	Dowling et al. (2012)	All High-Risk
APER	0.2587	0.2379	0.3922	0.6766
Precision	0.5846	0.6950	0.5000	
Recall	0.6032	0.6071	0.2625	0.3234
F Measure	0.5882	0.6754	0.4234	

Figure 20. Hybrid Classification Validation Results

Section Summary

We began this section by providing the results from our three model-building processes: forward stepwise discriminant analysis, backward stepwise discriminant analysis, and modified RGSS. Following this, we showed the performance of the best performing model against a withheld validation set and a LOOCV method. Finally, we conclude this section with a comparison across analysis methods using validated models from the multivariate classification method, multinomial Naïve Bayes method, and the Hybrid multivariate classification methods. In Figure 21, we see a two models tie for best performance overall, expressing identical performance. The Multinomial Naïve Bayes Classifier and the Hybrid Classifier dominate over every measure against all other models when considering long-term performance using LOOCV. We discuss this further and provide possible causes for this phenomenon in Chapter V. In the next section, we detail

the results of applying the methods used here to identify newly defined high-risk programs.



	Multivariate Classifier (LOOCV)	Modifed Dowling (2012)	Naïve Bayes Classifier (LOOCV)	Modifed Miller (2012)	Hybrid Classifier (LOOCV)	Modified Dowling et al. (2012)	All High-Risk
Accuracy	0.7242	0.6765	0.7621	0.6324	0.7621	0.3922	0.2919
Precision	0.4353	0.2143	0.6950	0.2381	0.6950	0.5000	
Recall	0.5823	0.2143	0.6071	0.1887	0.6071	0.2625	0.2919
F measure	0.5455	0.2143	0.6229	0.1969	0.6229	0.2901	

Figure 21. Validated Model Comparison Across Analysis Methods

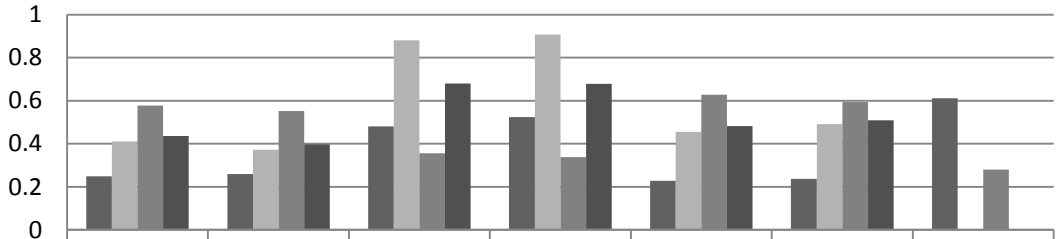
6-month Risk Models (Cumulative Change of Greater Than 5%)

In the previous section, we detailed our findings concerning the detection of programs at risk of experiencing a 6-month cumulative change in EAC of greater than 5% in magnitude. Here, we provide the results of our analysis for identifying programs at risk of experiencing a 6-month cumulative change of greater than 5%. Meaning, we only consider the negative consequences of cost growth as problematic as opposed to some magnitude of change (i.e. we ignore under budget programs).We first consider the multivariate classification methods. We again transition to the results of the multinomial Naïve Bayes classifier and display the results from our hybrid classification model. We conclude with a comparison across methods showing the best performing analysis method.

Multivariate Classification Results

We begin our multivariate classification results by comparing the top performing models from our three model building processes. Again, we use forward stepwise discriminant analysis, backward stepwise discriminant analysis, and modified RGSS. We follow this comparison by providing the results from our validation methods using both the withheld validation data and LOOCV.

In this multivariate classification analysis, we again seek models that provide the lowest APER. As shown in Figure 22, Modified RGSS1 again provides the best results relative to the other models, with modified RGSS2's performance providing a close second best. This suggests the modified RGSS method continues to search beyond the forward and backward stepwise discriminant analysis methods to find optimal solutions. We accomplish this by finding different statistically significant combinations of variables. In Table 15, we provide the composition of each model displayed in Figure 22. The models share many variables but the APERs vary widely.



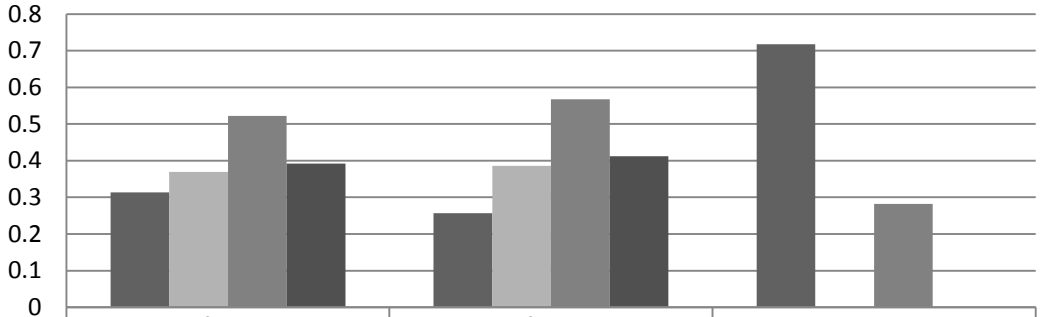
	Forward Stepwise1	Forward Stepwise2	Backward Stepwise1	Backward Stepwise2	Modified RGSS1	Modified RGSS2	All High-Risk
APER	0.2488	0.2599	0.4802	0.5235	0.2277	0.2364	0.6117
Precision	0.4115	0.3717	0.8805	0.9071	0.4558	0.4912	
Recall	0.5776	0.5526	0.3554	0.3377	0.6280	0.5936	0.2797
F measure	0.4366	0.3977	0.6796	0.6784	0.4822	0.5087	

Figure 22. Multivariate Classification Model Comparison

Table 15. Multivariate Classification Model Composition

	Forward Stepwise1	Forward Stepwise2	Backward Stepwise1	Backward Stepwise2	Modified RGSS1	Modified RGSS2
Generation					14	14
Iterations	11	10	25	26	232	233
P-Value to Enter	0.039873165	0.015267149	0.01271086	0.03486467	1.76825E-08	0.002508005
P-Value to Remove	0.016175135	0.023174606	0.01292904	0.01271086	0.014721308	0.009749843
APER	0.248762376	0.25990099	0.48019802	0.523514851	0.227722772	0.236386139
Precision	0.411504425	0.371681416	0.880530973	0.907079646	0.455752212	0.491150442
Recall	0.577639752	0.552631579	0.355357143	0.337726524	0.62804878	0.593582888
F measure	0.436619718	0.397727273	0.679644809	0.678358703	0.482209738	0.508707608
Variable count	11	10	15	16	10	11
Variables	SPI	SPI	TSPI	TSPI	SCI	SPI
	CV%	CV%	CV%	CV%	% Difference Between ML and W	SCI
	% Difference Between ML and W	% Difference Between ML and B	% Difference Between ML and B	% Difference Between ML and B	% Difference Between W and B	% Difference Between ML and W
	% Difference Between ML and B	CPI 1 Month Change	StDev CPI	StDev CPI	SCI StDev	% Difference Between W and B
	CPI 1 Month Change	TSPI 2 Month Change	CV% StDev	TCPI StDev	CV% StDev	SCI StDev
	TSPI 2 Month Change	Joint	CPI 2 Month Change	CV% StDev	CPI 1 Month Change	CV% StDev
	Joint	Comm.	CV% 2 Month Change	CPI 2 Month Change	TSPI 2 Month Change	CPI 1 Month Change
	Comm.	Plane	Joint	CV% 2 Month Change	Comm.	TSPI 2 Month Change
	Plane	Radar	Comm.	Joint	Radar	Comm.
	Radar	Small	Facility	Comm.	Small	Radar
	Small		Ship	Facility		Small
			Plane	Ship		
			Radar	Plane		
			Satellite	Radar		
			Small	Satellite		
				Small		

We executed our two-method validation on the Modified RGSS1 model. From Figure 23, we see that both validation methods provide relatively close results, but performs slightly worse than the training set. This may evidence over fitting in the training set, but we see the validation sets show relatively stable performance over several measures.



	Modifed RGSS1 (Withhold)	Modifed RGSS1 (LOOCV)	All High-Risk
APER	0.3134	0.2567	0.717542121
Precision	0.3692	0.3860	
Recall	0.5217	0.5670	0.2825
F Meaures	0.3921	0.4123	

Figure 23. Multivariate Validation

Multinomial Naïve Bayes Classifier Results

We begin detailing our results from our multinomial Naïve Bayes analysis by showing the effect of our add-α smoothing and MI thresholds on the error rate and vocabulary size. We found no comparable models and provide the simple untrained baseline model seen throughout our analysis for comparison purposes. As we saw from our previous analysis in identifying programs at risk for a change in the EAC of 5% or greater in magnitude, the α-level and MI thresholds strongly influenced the error rate. We see from Figures 24 and 25 that these models prove less sensitive to this effect when evaluating the α-level and MI threshold. For example, in Figure 11 of our prior multinomial Naïve Bayes classifier, we saw a 26% reduction of the average error rate as the MI increased. However, in Figure 25 we see a 16% reduction of the average error rate from the maximum to minimum levels.

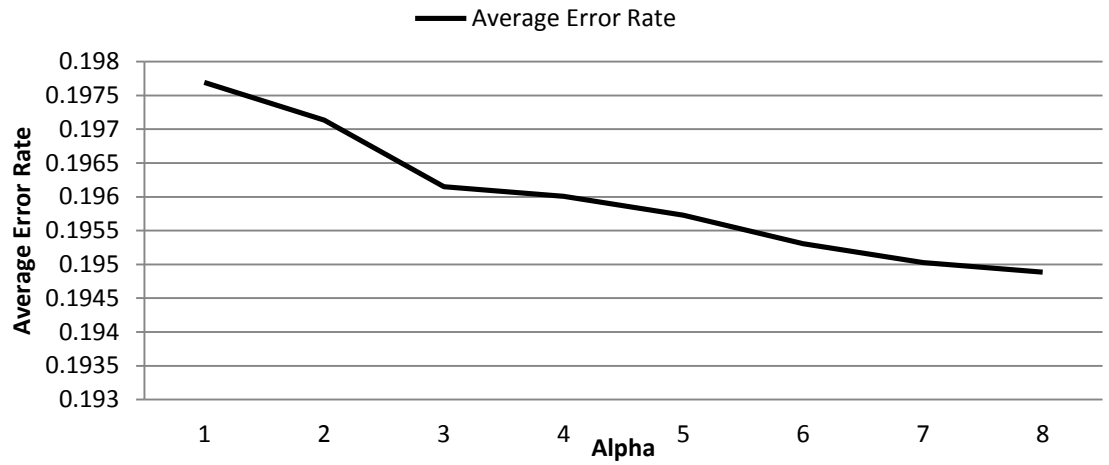


Figure 24. Average Error Rate vs. Add-α Smoothing level for Naïve Bayes text classification, 6-month cumulative change in EAC greater than 5% (The x-axis shown as qualitative scaling evaluated at $\alpha = \left(\frac{1}{4}^{i-1}\right), i = 1, \ldots, 8$

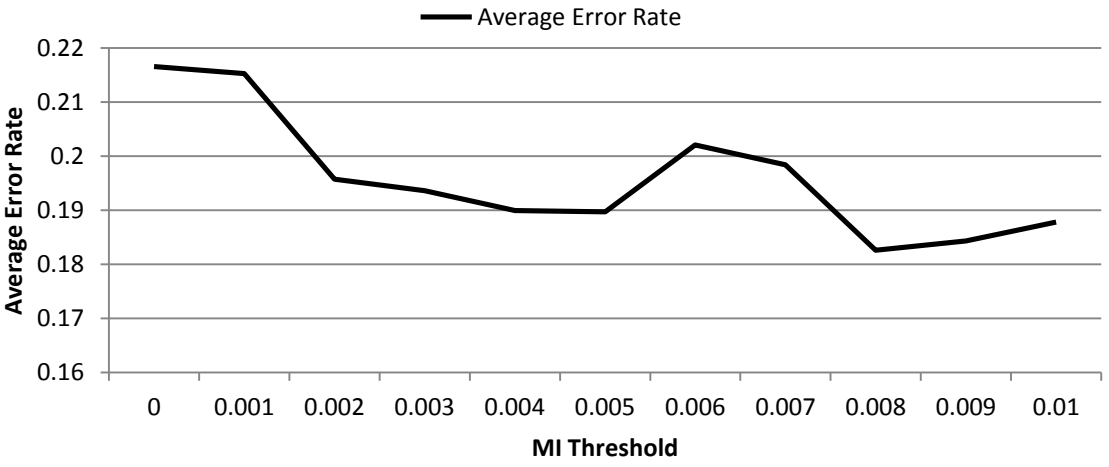
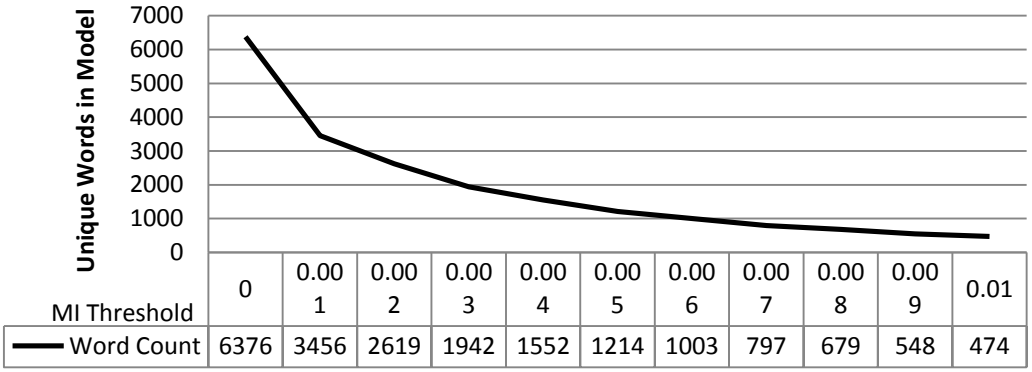


Figure 25. Average Error Rate vs. MI Threshold for Naïve Bayes text classification, 6-month cumulative change in EAC greater than 5%

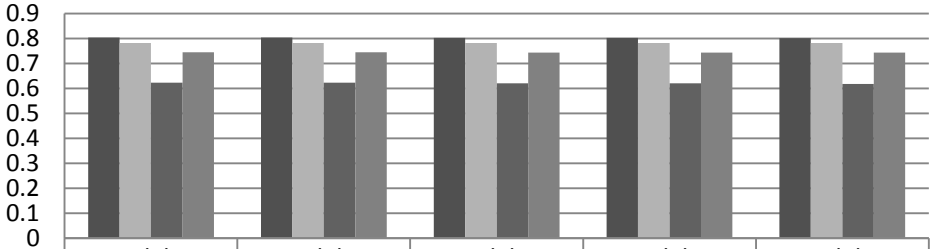
When comparing MI Threshold and Word Count, we find in Figure 26 a smaller vocabulary included in the analysis for this definition of high-risk over those found in our prior analysis. This seems to imply more ambiguity among the words in the high-risk category and nominal risk category due to a more restrictive definition of high-risk.



MI Threshold	0	0.001	0.002	0.003	0.004	0.005	0.006	0.007	0.008	0.009	0.01
Word Count	6376	3456	2619	1942	1552	1214	1003	797	679	548	474

Figure 26. Model Vocabulary vs. MI Threshold represents decreasing word count as MI increases

We evaluated 88 different models with differing levels of MI threshold and add-α smoothing. Figure 27 shows the top five performing models when measured by F measure. We see identical performance from models 63 and 64. In this case, we selected the model with the lower α-level. We see a 0.2% F measure performance difference between the highest performing model, Model 63, and lowest performing model, Model 58.



	Model 63	Model 64	Model 61	Model 62	Model 58
Accuracy	0.8037	0.8037	0.8022	0.8022	0.8006
Precision	0.7826	0.7826	0.7826	0.7826	0.7826
Recall	0.6234	0.6234	0.6207	0.6207	0.6180
F Measure	0.7446	0.7446	0.7438	0.7438	0.7430

Figure 27. Multinomial Naive Bayes Text Classifier Model Comparison

While the training data shows good performance, we must look to the validation results to understand how we expect the model to perform on new data. In concluding the

multinomial Naïve Bayes classifier portion of this section, we provide Figures 28 and 29 to show two aspects of our model's performance. First, in Figure 28, we see a reduction in words as more data becomes available. This seems to imply as more data and its true classification become available, we better differentiate words as important or not based on the mutual information provided by the word. Secondly, in Figure 29, we see an increase in performance as more data becomes available in the multiple stages of development.

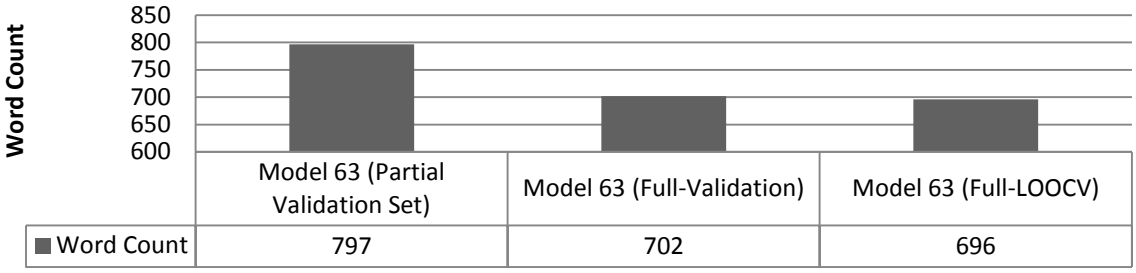
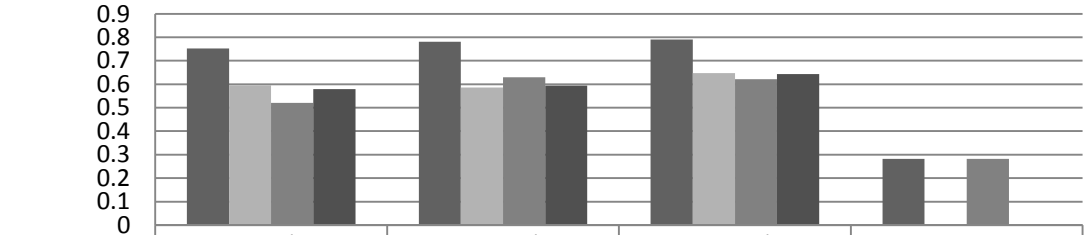


Figure 28. Vocabulary Learning



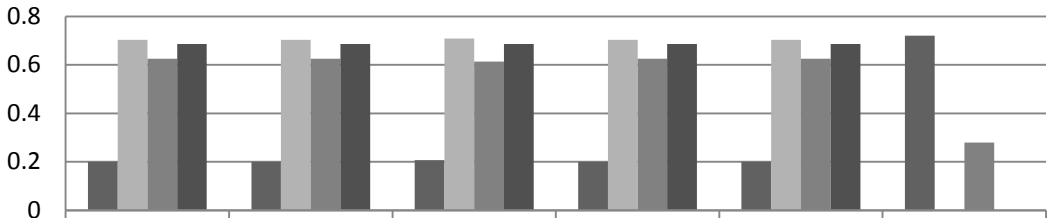
	Model 63 (Partial Validation)	Model 63 (Full-Withhold)	Model 63 (Full-LOOCV)	All High-Risk
Accuracy	0.7516	0.7811	0.7899	0.2815
Precision	0.5952	0.5862	0.6479	
Recall	0.5208	0.6296	0.6216	0.2815
F measure	0.5787	0.5944	0.6425	

Figure 29. Multi-Stage Validation

Hybrid Multivariate and Naïve Bayes Text Classification Model

In this section, we provide the result of our efforts to combine the multivariate classification and multinomial Naïve Bayes classifier to form a hybrid classification mod-

el. In our prior definition of high-risk programs, we showed the hybrid model development produced a tight grouping of performance measures for each of our three model building methods. Similarly, we see in Figure 30, the hybrid model development again produces a tight grouping of performance measures for our new definition of high-risk programs (those that experience a 6-month cumulative change in EAC greater than 5%). Interestingly, we see identical performance for four of the five models displayed. In Table 16, we provide variable composition for each model. We see from this table all five models contain three variables in common and deviate very little in selecting highly predictive variables. Additionally, when we compare the average number of variables required for the multivariate classification method displayed in Table 15, we find a 67% reduction in the average number of variables required to produce a predictive model. We also see a 72% reduction in the APER and 123% improvement in our ability accurately identify high-risk programs over the All High-Risk model.



	Forward Stepwise1	Forward Stepwise2	Backward Stepwise1	Modified RGSS1	Modified RGSS2	All High-Risk
APER	0.2005	0.2005	0.2067	0.2005	0.2005	0.7203
Precision	0.7035	0.7035	0.7080	0.7035	0.7035	
Recall	0.6260	0.6260	0.6130	0.6260	0.6260	0.2797
F measure	0.6865	0.6865	0.6867	0.6865	0.6865	

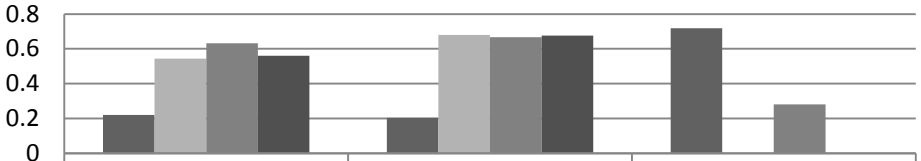
Figure 30. Hybrid Classifier Model Comparison

Table 16. Hybrid Classifier Model Output

	Forward Stepwise1	Forward Stepwise2	Backward Stepwise1	Modified RGSS1	Modified RGSS2
Generation				2	2
Iterations	3	4	35	36	37
P-Value to Enter	0.021865126	0.020855237	0.025285243	0.021865126	0.020855237
P-Value to Remove	0.004219305	0.021865126	0.019577328	0.004219305	0.021865126
APER	0.200495050	0.200495050	0.206683168	0.200495050	0.200495050
Precision	0.703539823	0.703539823	0.707964602	0.703539823	0.703539823
Recall	0.625984252	0.625984252	0.613026820	0.625984252	0.625984252
F measure	0.686528497	0.686528497	0.686695279	0.686528497	0.686528497
Variable Count	3	4	6	3	4
Variables	TSPI	TSPI	TSPI	TSPI	TSPI
	CV%	CV%	CV%	CV%	CV%
	NB_Pred_Class	Small	StDev CPI	NB_Pred_Class	Small
		NB_Pred_Class	CV% StDev		NB_Pred_Class
			Joint		
			NB_Pred_Class		

As mentioned in Chapter III, the lowest APER serves as our decision criteria for model selection in the multivariate classification and hybrid classification methods. We see identical performances from four models and see only two unique potential models, those represented by Forward Stepwise1, Modified RGSS1, and Forward Stepwise2, Modified RGSS2. We select the most parsimonious model for validation, resulting in the selection of Forward Stepwise1, or equivalently the Modified RGSS1.

Next, we applied our model to the withheld validation data and concluded our validation by performing LOOCV. As Figure 31 shows, in comparison with the withheld validation data, our LOOCV experienced a performance improvement in both Precision and Recall. We show a 24% improvement in Precision and a 5% improvement in Recall. This suggests an improved ability to lower false negative detections within the high-risk class. When compared to the simple All High-Risk classification rule we see a 137% improvement in our ability to identify correctly programs at risk of increasing costs.

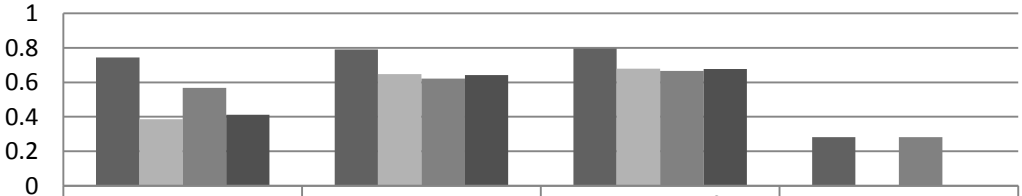


	Forward Stepwise1 (Withhold)	Forward Stepwise1 (LOOCV)	All High-Risk
APER	0.2200	0.2032	0.7185
Precision	0.5439	0.6790	
Recall	0.6327	0.6662	0.2815
F measure	0.5596	0.6764	

Figure 31. Hybrid Classification Validation Results

Section Summary

Much like the 6-month risk model seeking a cumulative change in EAC of greater than 5% in magnitude, we applied our multivariate classification method, multinomial Naïve Bayes classifier, and constructed a hybrid classification model. We conclude this section by providing a comparison across analysis methods using the LOOCV models from each method. As seen in Figure 32, the Hybrid classifier outperforms the multivariate classifier, the multinomial Naïve Bayes classifier, and simple untrained All High-Risk classification rule. We discuss possible causes for this further in Chapter V.



	Multivariate Classifier (LOOCV)	Naïve Bayes Classifier (LOOCV)	Hybrid Classifier (LOOCV)	All High-Risk
Accuracy	0.7433	0.7899	0.7968	0.2815
Precision	0.3860	0.6479	0.6790	
Recall	0.5670	0.6216	0.6662	0.2815
F Meaures	0.4123	0.6425	0.6764	

Figure 32. Validated Model Comparison Across Analysis Methods

12-month Risk Models (Cumulative Change of Greater Than 5%)

After evaluating our data for a 6-month cumulative change in EAC of greater than 5% in magnitude and then focusing on only the positive cumulative change of greater than 5%, we extend the effective time horizon of our model from 6-months to 12-months. We accomplish this by performing the same analysis on our dataset as outlined in our two previous definitions. We begin by outlining our results from the multivariate classification model. Then, we transition to the multinomial Naïve Bayes classifier. Finally, we provide our results for a hybrid model of the multivariate classification and multinomial Naïve Bayes classifier. We conclude by providing a cross-method comparison of performance for each validated model.

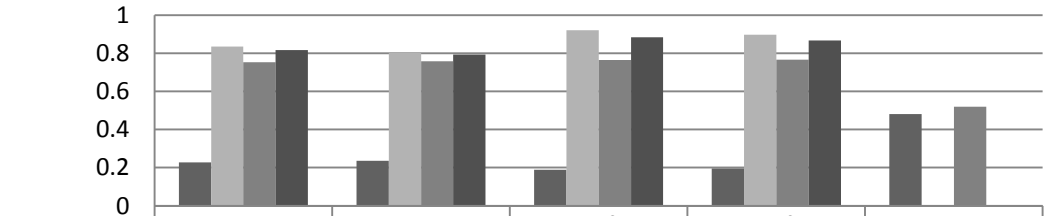
Multivariate Classification

We begin this section by providing a comparison of our top two performing models from each of our three model building processes. We note here, the backward stepwise discriminant analysis provided a list of statistically significant variables to evaluate. However, due to a limitation with the multivariate classification rule we cannot evaluate the proposed model.

In our 12-month model, the backward stepwise discriminant analysis found a list of statistically significant variables that provide good separation, but our classification rule proved unsuitable for evaluating these variables. We trace this problem to the covariance matrices for each of the two classes and the evaluation of Equation 14. In Equation 14, we evaluate the equation $k = \frac{1}{2}\ln\left(\frac{|\boldsymbol{S_1}|}{|\boldsymbol{S_2}|}\right) + \frac{1}{2}(\boldsymbol{\mu}_1'\mathbf{S}_1^{-1}\boldsymbol{\mu_1} - \boldsymbol{\mu}_2'\mathbf{S}_2^{-1}\boldsymbol{\mu_2})$. In our backward stepwise discriminant analysis, our model building process found variables that

produced a positive determinant of the covariance matrix for the nominal risk class and a negative determinant for the covariance matrix of the high-risk class. In evaluating for k, we attempt to find the natural log of a negative quotient, which results in an error. This means, potentially, we failed to evaluate a highly effective model. However, we argue our modified RGSS model building process proved its ability to search out potential models and return the highest performing models thus reducing the impact of this potential limitation. In our recommendations for future research, we provide other methods to overcome this limitation.

As shown in Figure 33, we see a sharp increase in our ability to identify correctly a program at risk of a 12-month cumulative change in the EAC of greater than 5%. Again, we see the Modified RGSS1 provides the best-performing model, showing a 60% reduction in APER when directly compared with the All High-Risk classification rule.



	Forward Stepwise1	Forward Stepwise2	Modified RGSS1	Modified RGSS2	All High-Risk
APER	0.2282	0.2358	0.1884	0.1960	0.4809
Precision	0.8348	0.8024	0.9204	0.8968	
Recall	0.7527	0.7577	0.7647	0.7657	0.5191
F measure	0.8170	0.7930	0.8844	0.8671	

Figure 33. Potential Multivariate Classification Model Comparison

In each of our model building methods, we evaluate the contribution a potential variable provides to discriminating between the two classes, high-risk and nominal risk. We see in Table 17, several variables present themselves in all of the models considered.

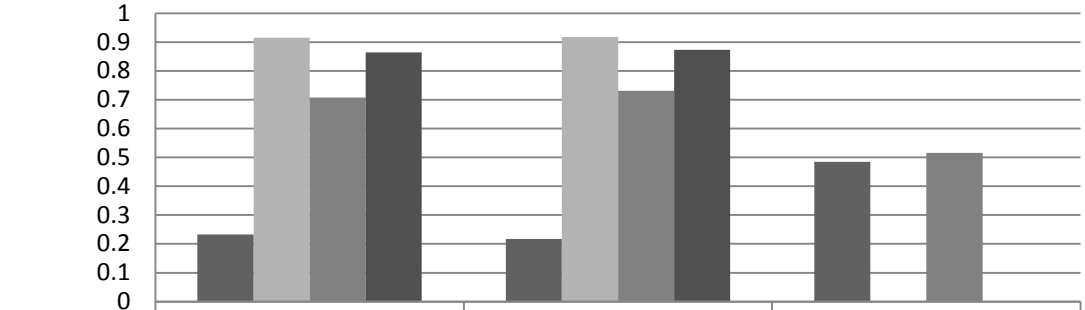
Unlike previous models considered in our prior definitions of high-risk programs, we see our best performing model consists of a large proportion of indicator variables for program type.

Table 17. Multivariate Classification Model Output

	Forward Stepwise1	Forward Stepwise2	Modified RGSS1	Modified RGSS2
Generation			22	22
Iterations	10	8	437	435
P-Value to Enter	0.003948007	0.000420858	0.02518918	0.012084581
P-Value to Remove	0.000561033	0.00028256	0.010099399	0.020989368
APER	0.228177642	0.235834609	0.188361409	0.196018377
Precision	0.83480826	0.802359882	0.920353982	0.896755162
Recall	0.752659574	0.757660167	0.764705882	0.765743073
F measure	0.816974596	0.793002915	0.884353741	0.867084997
Variable Count	8	8	15	15
Variables	SPI	CV%	% Complete	CPI
	CV%	% Difference Between ML and W	CPI	% Difference Between ML and W
	% Difference Between ML and W	% Difference Between W and B	% Difference Between ML and W	% Difference Between W and B
	% Difference Between W and B	StDev SPI	% Difference Between W and B	TCPI StDev
	TCPI StDev	TCPI StDev	TCPI StDev	SCI StDev
	Comm.	Comm.	SCI StDev	CV% StDev
	Radar	Radar	CV% StDev	SPI 1 Month Change
	Small	Small	AF	AF
			Comm.	Comm.
			Helicopter	Helicopter
			Ship	Ship
			Plane	Plane
			Radar	Radar
			Satellite	Satellite
			Small	Small

After selecting Modified RGSS1 as the best performing model, we look to our validation methods to evaluate the expected performance of our selected model against new data. From Figure 34, we see extremely high levels of Precision and Recall relative to the measures seen in prior sections of this chapter. Additionally, we see a strong trend

towards consistency of the performance measures when we compare the performance of the model on the validation sets and training set.



	Modified RGSS1 (Withhold Validation)	Modified RGSS1 (LOOCV)	All High-Risk
APER	0.2331	0.2169	0.4841
Precision	0.9146	0.9169	
Recall	0.7075	0.7311	0.5159
F-measure	0.8641	0.8725	

Figure 34. Multivariate Classification Validation Performance

Multinomial Naïve Bayes Classifier Results

Each month, via the CPR, contractors provide the DOD Format 5 data, which provides detailed descriptions of variances in cost and schedule as the program progresses. In this section, we present our results from using these Format 5s to identify programs at risk of experiencing a cumulative 12-month increase in EAC of greater than 5%. As seen from the previous multinomial Naïve Bayes classifier sections, add-α smoothing, but more specifically, MI thresholds tend to produce a strong influence on the model performance. We show in Figures 35, 36, and 37 the impact of add-α smoothing and MI thresholds influence this model's performance as well. Figure 35, shows a muted influence on the average error rate as we decrease the α-level. In previous sections, we

saw the average error rate decrease by more than 1% from the highest α-level to lowest; however, here we see the error rate falls less than 1% and quickly levels off.

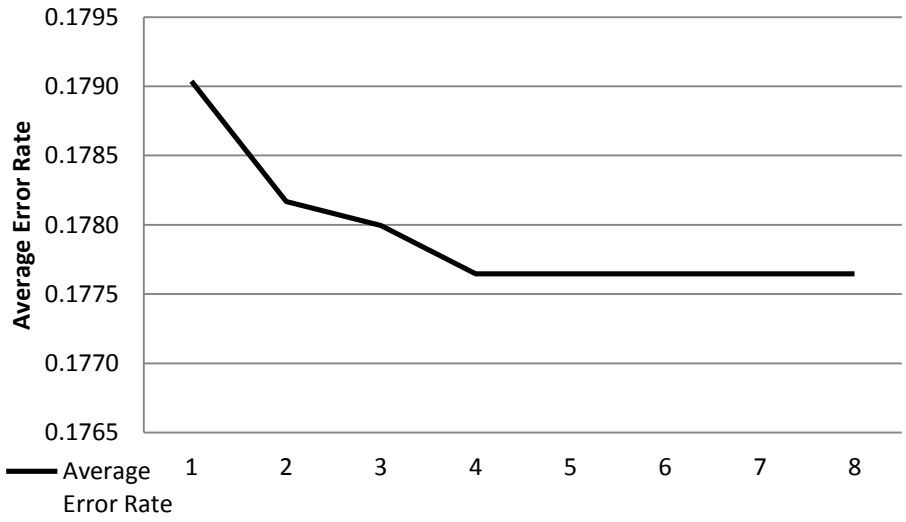


Figure 35. Average Error Rate vs. Add-α Smoothing level for Naïve Bayes text classification, 12-month cumulative change in EAC greater than 5% (The x-axis shown as qualitative scaling evaluated at $\boldsymbol{\alpha = \left(\frac{1}{4}^{i-1}\right), i = 1, \ldots, 8}$

When considering the MI thresholds impact on average error rate, we saw in Figures 14 and Figure 27 a decreasing downward trend for the average error rate as the MI threshold increased. However, using our current definition for high-risk programs, we see in Figure 37 a nearly parabolic shaped error rate. This seems to imply a greater sensitivity to the number of words included in our model vocabulary. We see in Figure 38, the number of words included meeting our MI threshold quickly decreases as the MI threshold increases. As the number of words decreases, Figure 37 implies the model vocabulary overfits the data.

As we previously discussed in Chapter III, overfitting occurs from "an incorrect generalization from an accidental property of the training set" (Manning, Raghavan, &

Schutze, 2008:251). In this case, we suggest one possible cause of the overfitting may result from dependencies between successive Format 5s from individual programs and poor generalization. In Figure 36, 12 monthly observations for a hypothetical program. In this simplistic example, we find the MI value relatively high due to the disproportionate number of observations in the high-risk class compared with the nominal risk class. In this example, the conditional probabilities equal and the prior probabilities of the classes dictate the classification decision. A lower MI threshold may provide more predictive words to the Naïve Bayes classifier. In the 6-month models, the shorter time period analyzed may reduce this effect. We relate this to Figure 37 by highlighting lower values for MI thresholds resulted in higher performing models.

	Format 5s											
Status	High-risk	High-risk	High-risk	High-risk	High-risk	High-risk	High-risk	High-risk	Nominal risk	Nominal risk	Nominal risk	Nominal risk
Word	1	2	3	4	5	6	7	8	9	10	11	12
Positive	1	1	1	1	1	1	1	1	1	0	0	1

Figure 36. Hypothetical program to illustrate potential cause of higher misclassification rates associated with higher Mutual Information thresholds in 12-month model building

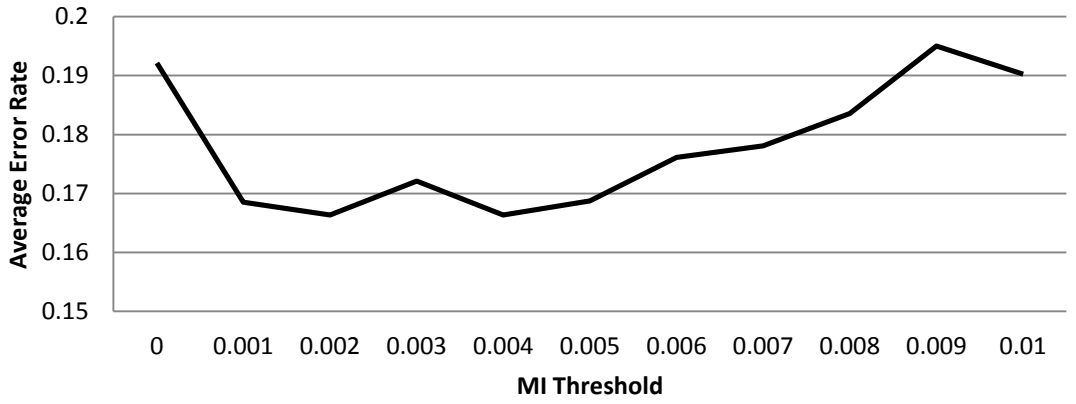


Figure 37. Average Error Rate vs. MI Threshold for Naïve Bayes text classification, 12-month cumulative change in EAC greater than 5%

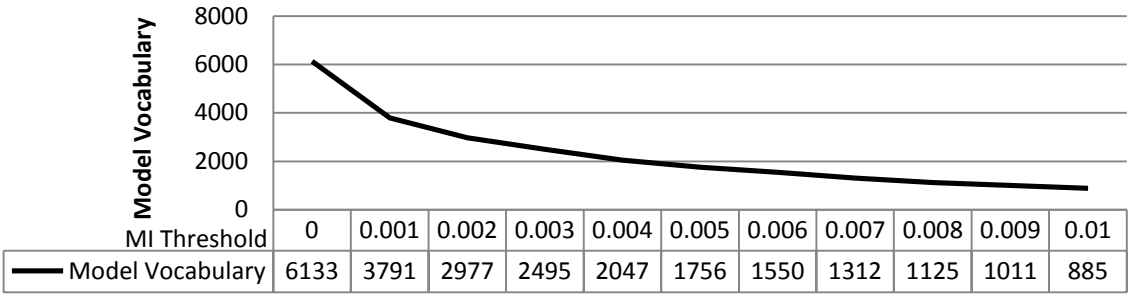
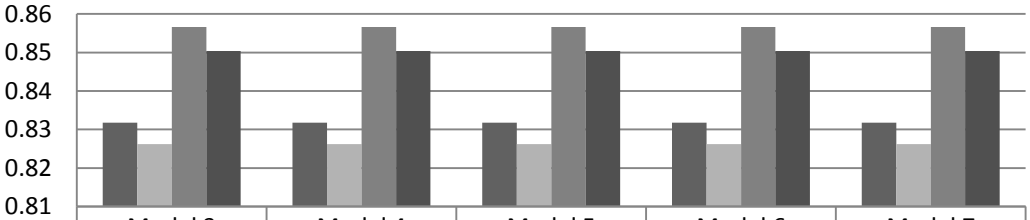


Figure 38. Model Vocabulary vs. MI Threshold represents decreasing word count as MI increases

From a list of 88 potential models evaluated at each level of MI threshold and α we have selected the top five performing models for display. As seen in Figure 39, the top five models showed equal performance across all performance measures and we selected the model with the lowest α level (see Table 18); this resulted in our selection of Model 7 for validation.

Table 18. 12-month Naive Bayes Top 5 performing models α-level

	Alpha Value
Model 3	0.015625
Model 4	0.00390625
Model 5	0.000976563
Model 6	0.000244141
Model 7	6.10352E-05



	Model 3	Model 4	Model 5	Model 6	Model 7
Accuracy	0.8317	0.8317	0.8317	0.8317	0.8317
Recall	0.8262	0.8262	0.8262	0.8262	0.8262
Precision	0.8566	0.8566	0.8566	0.8566	0.8566
F Measure	0.8504	0.8504	0.8504	0.8504	0.8504

Figure 39. Multinomial Naïve Bayes Text Classifier Model Comparison

As we validate our model, we seek to understand the impact of unseen data on the performance of the model. By evaluating our model using the multi-stage validation method we simulate the expected learning behavior of our model as new data becomes available. As seen in Figure 40, contrary to our previous multinomial Naïve Bayes classifiers, the number of words in the model vocabulary increases. This seems to imply that as data becomes available we find a greater concentration of words by class. This concentration provides an improved ability to differentiate between the high-risk programs and nominal risk programs. We find this supported by Figure 41, where we find the difference in Precision performance negligible between the Full Validation set and LOOCV, but we see a nearly 10% improvement in Recall.

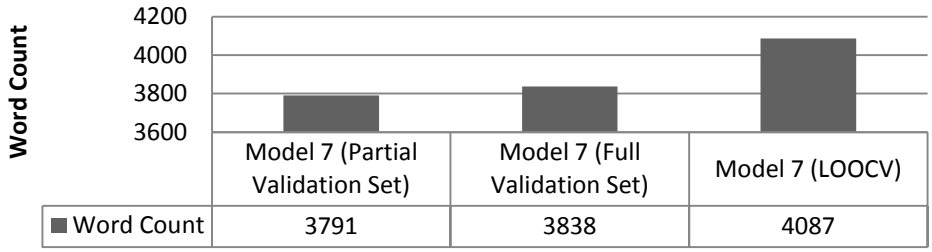
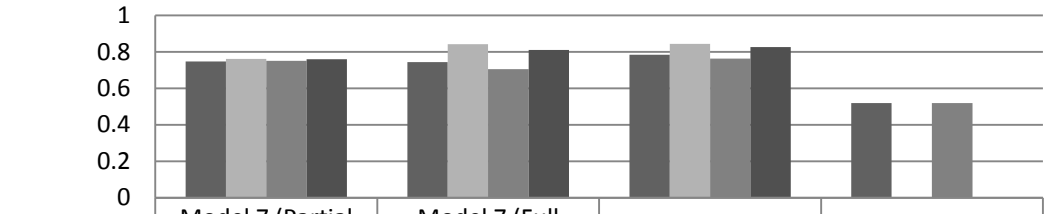


Figure 40. Vocabulary Learning

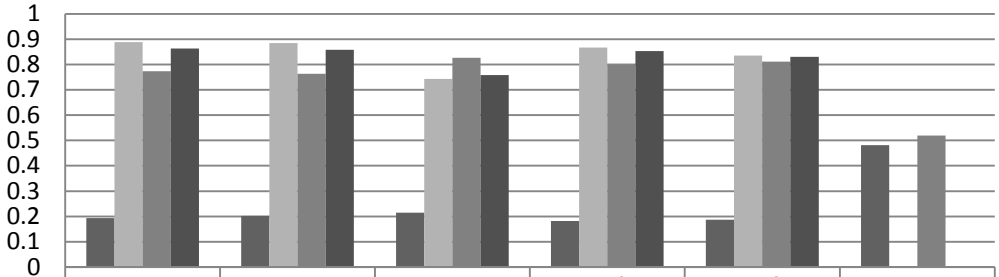


	Model 7 (Partial Validation Set)	Model 7 (Full Validation Set)	Model 7 (LOOCV)	All High-Risk
Accuracy	0.7462	0.7423	0.7831	0.5191
Precision	0.7612	0.8415	0.8432	
Recall	0.7500	0.7041	0.7618	0.5191
F Measure	0.7589	0.8099	0.8256	

Figure 41. Multi-Stage Validation

Hybrid Multivariate and Naïve Bayes Text Classification Model

In each of the prior high-risk definitions, we see the hybrid multivariate and Naïve Bayes classifier performs exceptionally well. As expected, the hybrid classification model showed strong performance and tight groupings of performance measures in each of our prior definitions of high-risk programs. We see in Figure 42, our current definition provides no exception from this trend. We draw the reader's attention to the performance difference between the All High-Risk classification rule and our best performing model measured by APER, the Modified RGSS1. We see a 53% improvement in Recall and a 62% reduction in the APER. In Table 19, we provide the variable composition for each model displayed in Figure 42. We see the variables *CV%*, *SCI*, *SPI*, and *% Difference Between ML and W*, as well as a handful of categorical variables repeat across models. This implies these variables provide good discriminatory power between the high-risk and nominal risk classes.



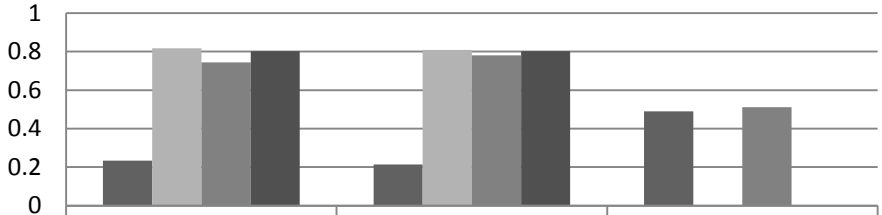
	Forward Stepwise1	Forward Stepwise2	Backward Stepwise1	Modified RGSS1	Modified RGSS2	All High-Risk
APER	0.1930	0.2021	0.2144	0.1822	0.1868	0.4809
Precision	0.8879	0.8850	0.7434	0.8673	0.8348	
Recall	0.7738	0.7634	0.8262	0.7989	0.8109	0.5191
F measure	0.8625	0.8576	0.7586	0.8527	0.8299	

Figure 42. Hybrid Classifier Model Comparison

Table 19. Hybrid Classifier Model Output

	Forward Stepwise1	Forward Stepwise2	Backward Stepwise1	Modified RGSS1	Modified RGSS2
Generation				2	18
Iterations	11	9	24	42	359
P-Value to Ente	0.020969915	0.014565773	0.042153999	0.029916791	0.036391816
P-Value to Rem	0.004082307	0.005933797	0.013215214	0.024260356	0.006332704
APER	0.19295559	0.202143951	0.2143951	0.182235835	0.186830015
Precision	0.887905605	0.884955752	0.743362832	0.867256637	0.83480826
Recall	0.77377892	0.763358779	0.826229508	0.798913043	0.810888252
F measure	0.862464183	0.857632933	0.758579169	0.852668213	0.829912023
Variable Count	11	9	17	11	11
Variables	SPI	SPI	TCPI	SPI	SCI
	CV%	CV%	SCI	SCI	CV%
	% Difference Between ML	% Difference Between ML	CV%	% Difference Between ML	% Difference Between ML
	% Difference Between ML	TCPI StDev	% Difference Between ML	SCI StDev	% Difference Between W
	TCPI StDev	Army	% Difference Between W	TCPI 2 Month Change	TCPI 2 Month Change
	Army	Plane	StDev CPI	Army	Army
	Comm.	Radar	TCPI StDev	Comm.	Comm.
	Plane	Small	CV% StDev	Plane	Plane
	Radar	NB_Pred_Class	TCPI 1 Month	Radar	Radar
	Small		CPI 2 Month	Small	Small
	NB_Pred_Class		CV% 2 Month	NB_Pred_Class	NB_Pred_Class
			Army		
			Comm.		
			Plane		
			Radar		
			Small		
			NB_Pred_Class		

From these models, we selected Modified RGSS1 as our top performing model following our multivariate model selection criteria of lowest APER. Next, we evaluated the withheld validation data by applying our Modified RGSS1. We provide the resulting observations in Figure 43 along with the results of applying the LOOCV method. From Figure 42 and Figure 43, we see a trend towards consistency in the performance of our model. This suggests that as additional data becomes available we expect the long-term performance of the model to remain stable. To reiterate, we seek lower values of APER and higher values for all other performance measures.



	Modified RGSS1 (Withhold)	Modified RGSS1 (LOOCV)	All High-Risk
APER	0.2331	0.2145	0.4890
Precision	0.8171	0.8082	
Recall	0.7444	0.7801	0.5110
F measure	0.8014	0.8024	

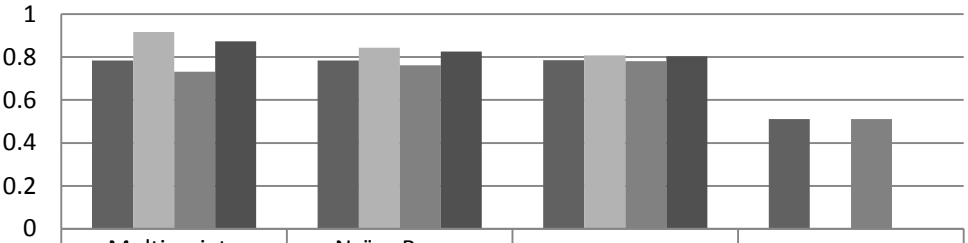
Figure 43. Hybrid Classification Validation Results

Section Summary

In our analysis of extending the effective timeframe of our model from 6-months to 12-months, we find the multivariate classifier outperforms both the hybrid classifier and multinomial Naïve Bayes classifier. We conclude this section by providing a comparison across analysis methods using the LOOCV models from each method. As seen in Figure 44, the Hybrid Classifier marginally outperforms the Naïve Bayes classifier when we measure by accuracy, but provides a slight performance edge in Recall. While the multivariate classifier performs exceptionally when measured by Precision, we find a 6% decline in Recall relative to the hybrid classifier.

During our analysis, we found the performance of the multivariate analysis potentially fit the data too well. To allay these concerns, we queried DCARC for data beyond that considered in our original database. We collected data on the Global Hawk program, a program not considered in our dataset, and applied the multivariate model to the data. We found similar results to that found in our dataset, we observed a Recall

measure of 88% and an overall accuracy of 89%. These new observations equated to less than 1% of our original dataset. Our model's performance on the Global Hawk data provided reassurances the model's performance did not result from an anomaly or overfitting the data.



	Multivariate Classifier (LOOCV)	Naïve Bayes Classifier (LOOCV)	Hybrid Classifier (LOOCV)	All High-Risk
Accuracy	0.7831	0.7831	0.7855	0.5110
Precision	0.9169	0.8432	0.8082	
Recall	0.7311	0.7618	0.7801	0.5110
F-measure	0.8725	0.8256	0.8024	

Figure 44. Validated Model Comparison Across Analysis Methods

Summary

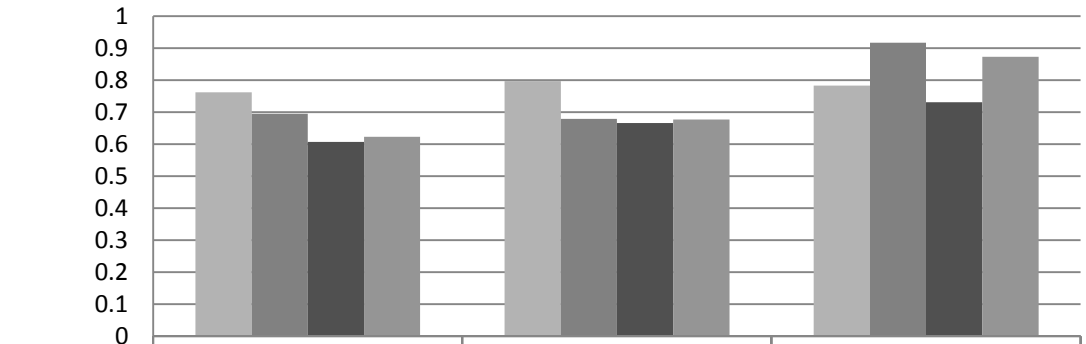
In this chapter, we presented our results beginning with our initial definition of high-risk programs, or those programs at risk of a 6-month cumulative change in EAC greater than 5% in magnitude. We then presented the results of our second high-risk program definition, or those programs at risk of a 6-month cumulative increase in EAC of greater than 5%. Finally, we presented the results for extending the effective detection window from a 6-month cumulative change in EAC greater than 5% to a 12-month cumulative change in EAC of greater than 5%.

In each definition of high-risk, we provided three methods for detecting high-risk programs: a multivariate classification method, a multinomial Naïve Bayes text classifier,

and a hybrid model joining both the multivariate and multinomial Naïve Bayes classifiers. In Figure 45, we provide the highest scoring models measured by F measure for each definition of a high-risk program. Definition 1 reflects the programs classified as high-risk if we see a 6-month cumulative EAC change greater than 5% in magnitude. Definition 2 corresponds to a 6-month cumulative increase in EAC of greater than 5%. Definition 3 represents the 12-month cumulative increase in the EAC of greater than 5%. In Figure 46, we provide conditional probability matrices for each of these definitions.

From the results of our analysis using Definition 2 and Definition 3, we observe a significant improvement by extending the time horizon. We believe one possible cause of this phenomenon arises from the length of time a new EAC takes to gain approval for reporting. Another possible cause relates to the existence of short-term reluctance that prevents rapid increases to the EAC, this may influence the short-term accuracy but have less of an effect on the long-term outcomes.

For the specific formulations of these models, we direct the readers to Appendix I, Appendix J, and Appendix K for the selected model in each definition respectively. In the next chapter, we discuss our findings further and relate them back to our original research questions. We also provide suggestions for improvement and direction for future research.



	Naïve Bayes Classifier (LOOCV) Definition 1	Hybrid Classifier (LOOCV) Definition 2	Multivariate Classifier (LOOCV) Definition 3
Accuracy	0.7621	0.7968	0.7831
Precision	0.6950	0.6790	0.9169
Recall	0.6071	0.6662	0.7311
F measure	0.6229	0.6764	0.8725

Figure 45. Selected Model for Each Definition of High-Risk

Definition 1: Naïve Bayes Classifier (LOOCV)			
		Predicted Class	
		High-Risk	Nominal Risk
Actual Class	High-Risk	0.6071	0.1504
	Nominal Risk	0.3929	0.8496
% of problems detected		69.50%	
% Accurate		76.21%	

Definition 2: Hybrid Classifier (LOOCV)			
		Predicted Class	
		High-Risk	Nominal Risk
Actual Class	High-Risk	0.6295	0.1307
	Nominal Risk	0.3705	0.8693
% of problems detected		67.90%	
% Accurate		79.68%	

Definition 3: Multivariate Classifier (LOOCV)			
		Predicted Class	
		High-Risk	Nominal Risk
Actual Class	High-Risk	0.7311	0.1215
	Nominal Risk	0.2689	0.8785
% of problems detected		91.69%	
% Accurate		78.31%	

Figure 46. Conditional Probability Matrices for best performing models in each definition of high-risk

V. Conclusions and Recommendations

Chapter Overview

In this chapter, we discuss the conclusions we draw from the results of our research effort and the implications this work may provide to the DOD's EVM and program management community. We also convey possible directions for further research and improvements to our methods. Finally, we state the significance of our study's findings to the EVM community, especially Cost Analysts and Program Managers.

Conclusions of Research

In Chapter I, we began our research by asking three questions:

1. Does adopting either a multivariate classification, multinomial Naïve Bayes text classifier, or a hybrid of the two methods, improve on prior methods used to identify programs at risk of a 6-month change in the EAC (either cost or under cost)?
2. If so, do these new methods allow us to identify programs at risk of cost growth greater than 5% 6-months out? 12-months out?
3. If we answer questions one and two affirmatively, can we incorporate these methods into tools available to the DOD program management community?

We begin our discussion by reflecting on question one. In Chapter IV, Figure 21 clearly shows the Naïve Bayes classifier and hybrid model outperforms all prior research

proxy models. We selected the Naïve Bayes classifier as the highest performing model in our evaluation of programs in the high-risk class defined by Definition 1 from Chapter IV. Our initial evaluation of the hybrid model in the training set showed promise but upon evaluation using LOOCV, we found the performance perfectly matched that of the Naïve Bayes classifier.

Digging in to the details of the hybrid classification method, we find in addition to *NB_Pred_Class* variable, only two variables from the Format 1 data that showed the required significance to gain entrance to the model (the *NB_Pred_Class* represents the predicted class provided by the Naïve Bayes classifier model). This seems to imply the Format 1 data does not contribute a significant amount of information over that already contributed by the variable *NB_Pred_Class*. The variable *NB_Pred_Class* dilutes any discriminating power provided by the Format 1 data as additional data enters the model and strengthens the discriminating power of the *NB_Pred_Class* variable. Given the convergence of models to identical performance measures, we select the most parsimonious model as best. In this case, we selected the Multinomial Naïve Bayes Classifier for identifying high-risk programs in our first research question.

Next, we consider our second research question. We see in Figure 45 of Chapter IV, two different models selected as best for each timeframe. First, we discuss the impact of changing the definition of high-risk programs from identifying the magnitude change to simply the cumulative change in EAC greater than 5%. We see Figure 45 displays the Hybrid Classifier as the best performing model when identifying high-risk programs using the new definition. The hybrid model consisted of two variables from the Format 1

data, we found *TSPI* and *CV%* highly discriminatory in addition to *NB_Pred_Class*. Interestingly, the two variables from Format 1 data focus on both schedule and cost aspects of performance respectively, not just cost.

Our second consideration in question two provides useful insight to the timeframe data remains effective. We see by extending our identification timeframe from 6-months to 12-months that we now select the multivariate classifier as the best performing model. We see in Chapter IV, Figure 45 shows the Multivariate classifier identifies 92% of the available high-risk programs in our dataset while providing a 73% chance of correctly identifying a program as high-risk. While we see a 4% decrease in Recall when compared to the Naïve Bayes classifier, we more than compensate for this loss by a nearly 9% increase in Precision. This suggests that over extended periods, the Format 1 data provides more useful information used to separate high-risk programs from nominal risk programs at a minimal cost of false detections. However, this does not suggest the Format 5 data provides no useful information, only that the multivariate classifier performed the best out of our alternatives.

Overall, our analysis using the cumulative change in EAC greater than 5% for the 6-month timeframe and 12-month timeframes, suggest that Format 5 data contributes more meaningfully to short term risk detection while the Format 1 data provides more significance over longer term risk detection.

Finally, we discuss our findings in relation to our third research question. Each of our analysis methods collected data currently available to DOD program management community. We purposefully avoided the use of commercial statistical software packages

requiring licensing. Instead, we completed this analysis using Microsoft Excel® and the free statistical software R (The R Foundation for Statistical Computing, 2011). We believe this methodology can provide significant advantage to the DOD program management community and ease of implementation.

Recommendations for Future Research

During the course of our research, we identified three promising avenues for further research. First, we discuss the potential use of the Naïve Bayes classifier to identify potential root causes of increased risk. Secondly, we discuss enhancements to our methodology that may provide additional insight to the identification of high-risk programs. Finally, we discuss methods to predict the specific cumulative EAC change expected for programs considered high-risk.

We believe the possibility exists, where we can apply the Naïve Bayes classifier to Selected Acquisition Reports (SARs) and map the cost variance descriptions to the CPR Format 5s. The SARs provide annual status updates for DOD MDAPs and contains eight cost variance categories (Department of Defense, 2011). We provide these eight categories in Table 20. If we treat each category as a class, we believe it is possible to train the Naïve Bayes classifier for key terms in each class and apply this model to the Format 5 data provided each month. By doing so, we provide some insight to possible root causes of risk within the program and provide Program Managers a more defined risk profile.

Table 20. SAR Cost Variance Categories (Department of Defense, 2011:19)

Economic	Schedule	Other
Quantity	Engineering	Support
Schedule	Estimating	

Our second recommendation for further research improves upon the methodology we set out in our analysis. In Chapter IV, pages 90 and 91, we mentioned the limitation associated with using our classification rule. One possible method to overcome the aforementioned limitation, we derive from the multivariate classification rule used by JMP®. This method uses the Mahalanobis distance and evaluates an observation's distance from the multivariate mean of each class (SAS Institute Inc, 2013b). The decision rule then chooses the class that minimizes this distance. During our evaluation of the multivariate classification method, we cross checked our analysis with that in JMP® and found the outputs in most cases identical. The exception relates to the limitation of the use of our probability density function. The Mahalanobis distance does not experience the same limitation and provides the additional advantage of providing a probability an observation belongs to the specific class predicted (SAS Institute Inc., 2013c).

In our multivariate classification, our classification rule treats the predicted class as certainty. We do not consider the probability of the observation belonging to a specific class during classification. By using the Mahalanobis distance, future research may provide higher Recall and Precision by setting detection thresholds on the probabilities provided for each observation. For example, in our multivariate analysis we may predict a nominal risk observation belongs to the high-risk class. However, the Mahalanobis

distance method may provide a probability the observation belongs to the high-risk class of 0.51. The research can set a threshold for the probability of greater than 0.55 before the detection method identifies a program as high-risk thus influencing the probabilities of false detections.

Our final recommendation involves a better method for predicting the actual change in the EAC using Ordinary Least Squares (OLS) Regression. White et al. (2004) used a two-step regression procedure to predict the amount of cost growth (as measured by Selected Acquisition Reports) an acquisition program would incur. Similarly, we suggest differentiating programs that express nominal risk profiles from those that express high-risk profiles using our methodology, then use multiple regression to predict the amount of cost growth expected from the high-risk population. This would further enhance our model's ability to provide useful input to the LCRM matrix discussed in our significance of research section and improve decision support.

Significance of Research

This research effort significantly contributes to the current body of knowledge on DOD acquisition risk detection and provides useful application for DOD program management. We previously showed the significant improvement in our ability to identify correctly, programs at risk of changes in EAC and the probabilities associated with these methods compared with prior research. We now discuss the additional significance of this effort.

We find this research effort not only provides an overall program risk identification but we can change the specific labels applied to the observations and change the learning objective. This becomes useful if we consider this in the context of risk reporting to program management. Air Force Pamphlet 63-128 provides guidance on assessing life cycle risk management (LCRM) (Department of the Air Force, 2009:107-109). This guidance provides specific direction on the development on commonly used LCRM Risk Matrices. Figure 47 provides a visual representation of this LCRM Risk Matrix.

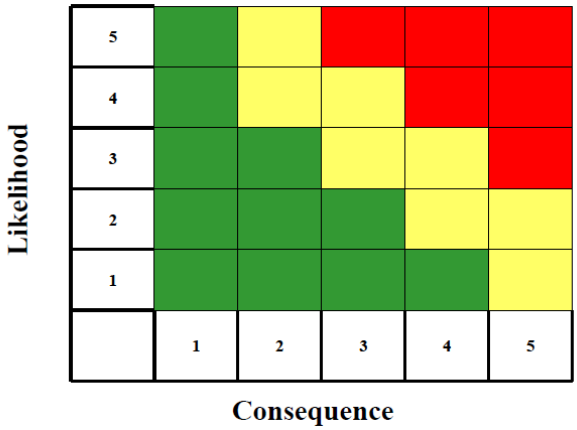


Figure 47. LCRM Risk Matrix (Department of the Air Force, 2009:107)

Additionally, we provide Table 21 and Table 22, which respectively provides the likelihood criteria and the cost consequence criteria used to determine visually risk of the program.

Table 21. Likelihood Criteria (Department of the Air Force, 2009:107)

Level	Likelihood	Probability of Occurrence
5	Near Certainty	81%-99%
4	Highly Likely	61%-80%
3	Likely	41%-60%
2	Low Likelihood	21%-40%
1	Not Likely	5%-20%

Table 22. Standard AF Consequence Criteria – Cost (Department of the Air Force, 2009:109)

LEVEL	Standard AF Consequence Criteria – Cost (A-B refers to MS)
1	For A-B Programs: 5% or less increase from MS A approved cost estimate For Post-B & Other Programs: limited to <=1% increase in Program Acquisition Unit Cost (PAUC) or Average Procurement Unit Cost (APUC) from current baseline estimate, or last approved program cost estimate
2	For A-B Programs: > 5% to 10% increase from MS A approved estimate For Post-B & Other Programs: <=1% increase in PAUC/APUC from current baseline estimate, or last approved program cost estimate, with potential for further cost increase
3	For A-B Programs: >10% to 15% increase from MS A approved estimate For Post-B & Other Programs: >1% but <5% increase in PAUC/APUC from current baseline estimate, or last approved program cost estimate
4	For A-B Programs: >15% to 20% increase from MS A approved estimate For Post-B & Other Programs: 5% but <10% increase in PAUC/APUC from current baseline estimate, or last approved program cost estimate
5	For A-B Programs: >20% increase from MS A approved cost estimate For Post-B & Other Programs: >=10% increase in PAUC/APUC from current baseline estimate (danger zone for significant cost growth and Nunn-McCurdy breach), or last approved program cost estimate

In the context of Table 21, our model currently provides the program analyst a level 4 output. In other words, if we identify the program as high-risk we see the probability falls within the definition of the highly likely category. Additionally, our model produces expected cost growth of the program overall (greater than 5%). We see this does not specifically match the criteria set out in Table 22, as these thresholds consider the PAUC/APUC. However, by changing our learning objective to higher cost growth the analysts can quickly determine the minimum expected cost growth for PAUC/APUC by dividing the new expected minimum EAC by the number of units in acquisition.

To clarify further we provide the following example. Consider a post-milestone B program acquiring a single unit of some product with an EAC of $100. We run our 6-

month analysis on the CPR provided by the contractor and find the program identified as high-risk. At a minimum, this implies we expect the program EAC to rise to $105 in 6-months. Accordingly, we look to report this development to program management using the LCRM risk matrix. We find our identification methods provide a level 4 probability of occurrence and we expect our APUC to increase by a minimum of 5%. We plot the risk profile for this program in coordinates (4,4), as seen in Figure 48.

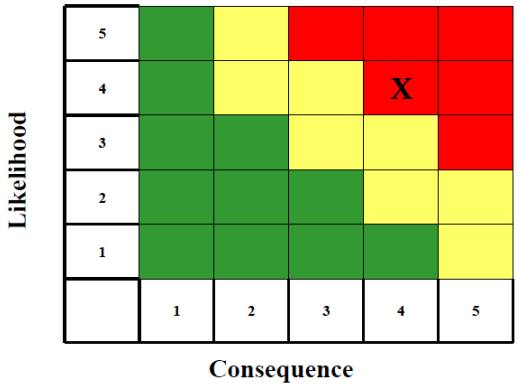


Figure 48. Example LCRM Risk Matrix Analysis

The flexibility of our learning methods allows us to define our learning objectives to match the criteria laid out in Table 22. This provides a probability of occurrence for each consequence level. In our recommendations for future research, we provided the potential for additional methods more appropriate for this type of analysis.

We conclude this research effort by providing details on opportunities for immediate implementation and integration into applications currently in use by the DOD acquisition community. We discuss three different opportunities. First, we discuss the CPR File Viewer. Next, we discuss the EVM-CR Dashboard. Finally, we discuss implementation in the *EVM_Analyst* role in DCARC.

We provide two alternatives for implementation of our model into the CPR File Viewer. In Chapter III, on page 31, we provided a passing comment on the fact that DCARC recently provided a CPR file viewer to overcome limitations associated with data collection in this and previous research. We see the CPR file viewer currently provides the option to highlight changes in CPI and SPI based on surpassing some user-defined threshold (Defense Cost and Reporting Center, 2013:9). We provide Figure 49 to illustrate this further.

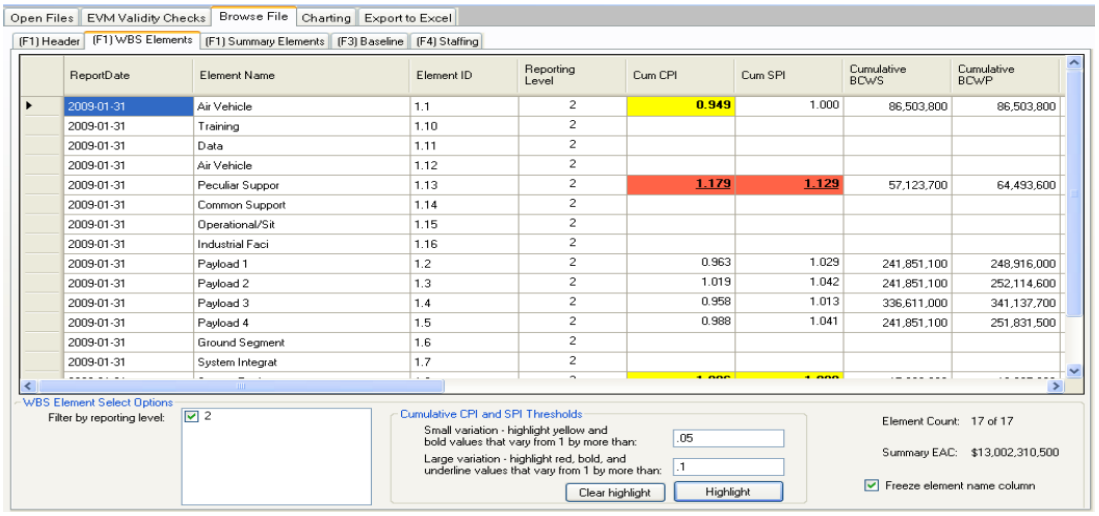


Figure 49. CPR File Viewer Risk Indicator (Defense Cost and Resource Center, 2013a:9)

We see these indicators as an interest in identifying increased risk to program cost performance. This interest presents an opportunity to integrate our model into an application currently in use and provide enhanced capabilities in program risk detection. The first alternative we discuss consists of integrating our model directly into the CPR File Viewer. Specifically, we suggest the model output alert users of the risk level of the program in *(F1) Header* tab of the *Browse File* tab (see Figure 50).

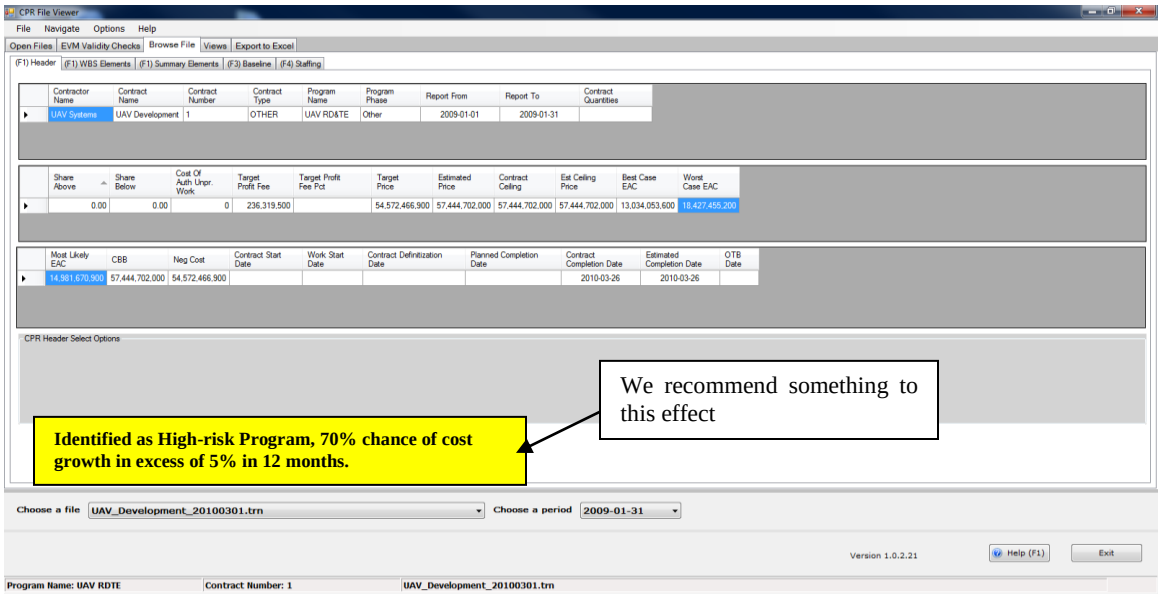


Figure 50. Recommended integration of the 12-month multivariate classification model to the EVM File Viewer

Our second alternative for the CPR File Viewer requires the addition of a *Risk Summary* tab. This tab would consist of the LCRM risk matrix from Figure 48. We also recommend including a modified version of Tables 21 and 22 specific to the probability and impact defined by DCARC. The user could then reference these tables for specific information about the LCRM risk matrix output.

The next recommendation we make, stems from our review of the DCARC EVM-CR dashboard, shown in Figure 51. We see the EVM-CR Dashboard implies the CPI and SPI provide sufficient information to gain perspective on the overall health of the DOD program portfolio. Based on an ad hoc analysis using our multivariate classification analysis for the 12-month increase in EAC of greater than 5%, we find the CPI and SPI indicate very little in identifying risk of potential cost growth given a specific timeframe. In contrast, our model provides a 30% improvement in overall accuracy in differentiating

between high-risk and nominal risk programs. We present the results of our ad hoc analysis in Figure 52 and provide results from our 12-month model in Figure 53 for comparison. We recommend adding an additional screen to the EVM-CR dashboard, which implements our model and identifies programs as high-risk or nominal risk. This provides more clarity and urgency to risks that CPI and SPI alone may not identify.

Conveniently, the office responsible for these applications also maintains the source data for our model. By providing analysts with the output from our model without the effort of conducting the analysis, we reduce the aversion to adopting new methodologies and ensure consistency of its application by maintaining the model in a single location.

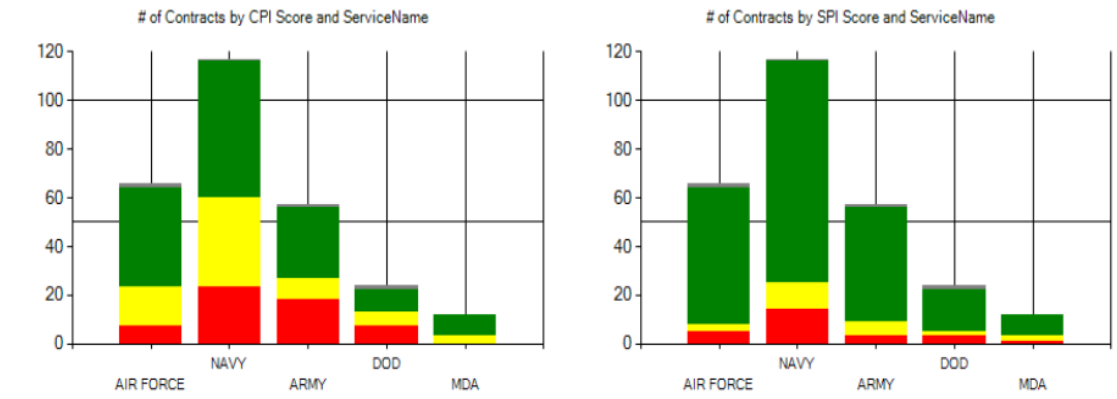


Figure 51. Screenshot of EVM-CR Dashboard showing CPI and SPI indicators (Defense Cost and Resource Center, 2013b)

SPI Only			
		Predicted Class	
		High-risk	Nominal Risk
Actual Class	High-risk	0.6579	0.4929
	Nominal Risk	0.3421	0.5071
% of problem detect		17.81%	
% Accurate		52.82%	

CPI Only			
		Predicted Class	
		High-risk	Nominal Risk
Actual Class	High-risk	0.5755	0.3155
	Nominal Risk	0.4245	0.6845
% of problem detect		85.99%	
% Accurate		60.05%	

CPI and SPI			
		Predicted Class	
		High-risk	Nominal Risk
Actual Class	High-risk	0.6004	0.4023
	Nominal Risk	0.3996	0.5977
% of problem detect		66.75%	
% Accurate		59.93%	

Figure 52. Ad hoc 12-month risk identification using only SPI and CPI for input

Definition 3: Multivariate Classifier (LOOCV)			
		Predicted Class	
		High-Risk	Nominal Risk
Actual Class	High-Risk	0.7311	0.1215
	Nominal Risk	0.2689	0.8785
% of problems detected		91.69%	
% Accurate		78.31%	

Figure 53. Multivariate Classifier (LOOCV) model seeking to identify programs at risk of 12-month cost growth greater than 5%

Our third and final recommendation applies directly to the *EVM_Analyst* role within DCARC. We see by selecting a program, the analyst can review program specific detail including a status assessment. Figure 54 illustrates this tool. We recommend the addition of a risk metric as shown in Figure 55. By providing the risk information on the program detail screen, we give analyst an advantage when synthesizing critical information concerning the overall health of the program.

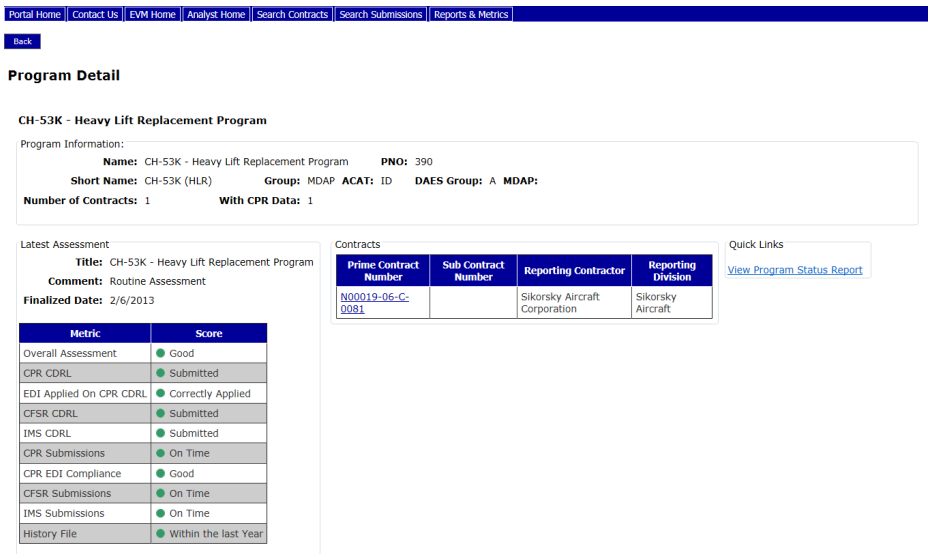


Figure 54. Screenshot from DCARC EVM_Analyst role program detail of CH-53K

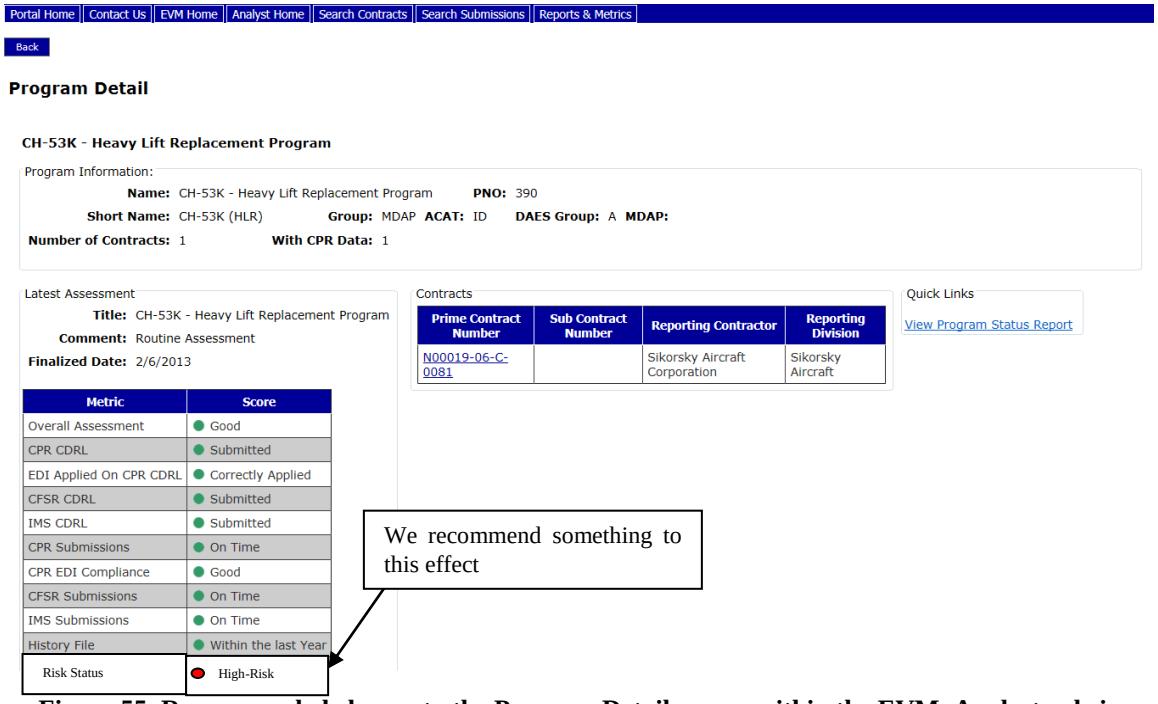


Figure 55. Recommended change to the Program Detail screen within the EVM_Analyst role in DCARC

This research effort reflects a culmination of three years of research seeking solutions to the problem of identifying programs with elevated levels of cost risk. Keaton et al. (2011) began by asking the question can the CPI and SPI provide the necessary information required to determine automate the detection of cost risk. They found the process showed potential for automation but their model proved too sensitive for implementation and resulted in too many false detections. Dowling (2012) adopted a more robust optimization technique to improve insight into the timeframe and probability of the occurrence of a cost risk. Miller (2012) asked does the Format 5 data provide any useful information in identifying programs with cost risks. In the same year, Dowling et

al. (2012) developed a unified model, bridging the gap between the analysis of Format 1 data and Format 5 data.

Yet despite Dowling (2012), Miller (2012), and Dowling et al. (2012) attempts, none of these provided an actionable decision support tool for the acquisition community. This research does that and more. The current research effort acts as the capstone, concentrating the knowledge collected from these previous efforts, improving upon these results, and providing an actionable decision support tool for the DOD acquisition community. We find this research directly supports the goals of "more disciplined use of resources" and "improving efficiency" laid out in the OUSD(Comptroller) FY2013 Defense Budget (Department of Defense, 2012a:3.1). Our research and methodology greatly aids in detecting high-risk programs in these cost-conscious times keeping program managers focused on the risk management horizon.

Appendix A: Variable List for Additional Data Calculations (Dowling, 2012)

Variable Name	Equation	Interpretation
6 Mo Delta	$6Mo\Delta\% = \frac{t_{i+6}-t_i}{t_i}$; where t_i is the EAC-ML for the ith month	Shows the 6 month change in EAC Most Likely
% Complete	$\%Complete = \left(\frac{BCWP_{CUM}}{BAC}\right) * 100$	Compares work accomplished to total work planned
Cost Performance Index (CPI)	$CPI = \frac{BCWP}{ACWP}$	Compares the budgeted cost for work completed against the actual cost of work completed
Schedule Performance Index (SPI)	$SPI = \frac{BCWP}{BCWS}$	Compares the budgeted cost for work performed against the budget cost of work scheduled
Total Schedule Performance Index (TSPI)	$TSPI = \frac{BAC - BCWP}{BAC - BCWS}$	Ratio of the budgeted performance to schedule performance
Total Cost Performance Index (TCPI)	$TCPI = \frac{BAC - BCWP}{EAC - ACWP}$	Ratio of budgeted performance to actual performance
Schedule Cost Index (SCI)	$SCI = CPI * SPI$	Cost ratio multiplied by schedule ratio
Schedule Variance (SV%)	$SV\% = \frac{BCWP - BCWS}{BCWS} * 100$	Shows ahead and behind schedule
Cost Variance (CV%)	$CV\% = \frac{BCWP - ACWP}{BCWP} * 100$	Shows over and under budget
% Difference Between W and ML	$\%Diff_{W-ML} = \frac{EAC_W - EAC_{ML}}{EAC_{ML}}$	The % difference between the contractor's worst case EAC estimate and the most likely EAC
% Difference Between ML and B	$\%Diff_{ML-B} = \frac{EAC_{ML} - EAC_B}{EAC_{ML}}$	The % difference between the contractor's most likely EAC estimate and the best case EAC
% Difference Between W and B	$\%Diff_{W-B} = \frac{EAC_W - EAC_{ML}}{EAC_W}$	The % difference between the contractor's worst case EAC estimate and the best case EAC

Variable Name	Equation	Interpretation
Standard Deviation CPI (StDev CPI)	$StDev(CPI) = StDev(CPI_t, CPI_{t-1}, CPI_{t-2})$	Measure of variability of the last three CPIs
Standard Deviation SPI (StDev SPI)	$StDev(SPI) = StDev(SPI_t, SPI_{t-1}, SPI_{t-2})$	Measure of variability of the last three SPIs
Standard Deviation TSPI (TSPI StDev)	$StDev(SPI) = StDev(SPI_t, SPI_{t-1}, SPI_{t-2})$	Measure of variability of the last three TSPIs
SCI Standard Deviation (SCI StDev)	$StDev(SCI) = StDev(SCI_t, SCI_{t-1}, SCI_{t-2})$	Measure of variability of the last three SCIs
Standard Deviation TCPI (TCPI StDev)	$StDev(TCPI) = StDev(TCPI_t, TCPI_{t-1}, TCPI_{t-2})$	Measure of variability of the last three TCPIs
SV% Standard Deviation (SV% StDev)	$StDev(SV\%) = StDev(SV\%_t, SV\%_{t-1}, SV\%_{t-2})$	Measure of variability of the last three SV%s
CV% Standard Deviation (CV% StDev)	$StDev(CV\%) = StDev(CV\%_t, CV\%_{t-1}, CV\%_{t-2})$	Measure of variability of the last three CV%s
CPI 1 Month Change	$CPI\ 1Mo\ \Delta\% = \frac{CPI_t - CPI_{t-1}}{CPI_{t-1}}$	Measure of the change from CPI of one month to the next month
SPI 1 Month Change	$SPI\ 1Mo\ \Delta\% = \frac{SPI_t - SPI_{t-1}}{SPI_{t-1}}$	Measure of the change from SPI of one month to the next month
TSPI 1 Month Change	$TSPI\ 1Mo\ \Delta\% = \frac{TSPI_t - TSPI_{t-1}}{TSPI_{t-1}}$	Measure of the change from TSPI of one month to the next month
TCPI 1 Month Change	$TCPI\ 1Mo\ \Delta\% = \frac{TCPI_t - TCPI_{t-1}}{TCPI_{t-1}}$	Measure of the change from TCPI of one month to the next month

Variable Name	Equation	Interpretation
SCI 1 Month Change	$SCI\ 1Mo\ \Delta\% = \frac{SCI_t - SCI_{t-1}}{SCI_{t-1}}$	Measure of the change from SCI of one month to the next month
SV% 1 Month Change	$SV\%\ 1Mo\ \Delta\% = \frac{SV\%_t - SV\%_{t-1}}{SV\%_{t-1}}$	Measure of the change from SV% of one month to the next month
CV% 1 Month Change	$CV\%\ 1Mo\ \Delta\% = \frac{CV\%_t - CV\%_{t-1}}{CV\%_{t-1}}$	Measure of the change from CV% of one month to the next month
CPI 2 Month Change	$CPI\ 2Mo\ \Delta\% = \frac{CPI_t - CPI_{t-2}}{CPI_{t-2}}$	Measure 2 month percent change in CPI
SPI 2 Month Change	$SPI\ 2Mo\ \Delta\% = \frac{SPI_t - SPI_{t-2}}{SPI_{t-2}}$	Measure 2 month percent change in SPI
TSPI 2 Month Change	$TSPI\ 2Mo\ \Delta\% = \frac{TSPI_t - TSPI_{t-2}}{TSPI_{t-2}}$	Measure 2 month percent change in TSPI
TCPI 2 Month Change	$TCPI\ 2Mo\ \Delta\% = \frac{TCPI_t - TCPI_{t-2}}{TCPI_{t-2}}$	Measure 2 month percent change in TCPI
SCI 2 Month Change	$SCI\ 2Mo\ \Delta\% = \frac{SCI_t - SCI_{t-2}}{SCI_{t-2}}$	Measure 2 month percent change in SCI
SV% 2 Month Change	$SV\%\ 2Mo\ \Delta\% = \frac{SV\%_t - SV\%_{t-2}}{SV\%_{t-2}}$	Measure 2 month percent change in SV%
CV% 2 Month Change	$CV\%\ 2Mo\ \Delta\% = \frac{CV\%_t - CV\%_{t-2}}{CV\%_{t-2}}$	Measure 2 month percent change in CV%

Appendix B: Perfect Correlation SPI and SV% Decomposition

The calculation for SPI follows this form:

$$\boldsymbol{SPI = \frac{BCWP}{BCWS}}$$

Next, we deconstruct the SV%, ignoring the multiplication of 100 as a constant, to find the SV% as follows:

$$\boldsymbol{SV}\% \propto \frac{\boldsymbol{SV}}{\boldsymbol{BCWS}} = \frac{\boldsymbol{BCWP - BCWS}}{\boldsymbol{BCWS}} = \frac{\boldsymbol{BCWP}}{\boldsymbol{BCWS}} - \frac{\boldsymbol{BCWS}}{\boldsymbol{BCWS}} = \frac{\boldsymbol{BCWP}}{\boldsymbol{BCWS}} - \mathbf{1} = \boldsymbol{SPI} - \mathbf{1}$$

Through this decomposition, we have shown SV% will always have a perfectly correlated relationship with SPI.

Appendix C: Variable List

Calculated Variables (See Appendix A for calculation details)

- % Complete
- CPI
- SPI
- TSPI
- TCPI
- SCI
- CV%
- % Difference Between ML and W
- % Difference Between ML and B
- % Difference Between W and B
- StDev CPI
- StDev SPI
- TSPI StDev
- TCPI StDev
- SCI StDev
- CV% StDev
- CPI 1 Month Change
- SPI 1 Month Change
- TSPI 1 Month Change
- TCPI 1 Month Change
- SCI 1 Month Change
- CV% 1 Month Change
- CPI 2 Month Change
- SPI 2 Month Change
- TSPI 2 Month Change
- TCPI 2 Month Change
- SCI 2 Month Change
- CV% 2 Month Change

Service

- AF
- Army
- Joint
- Navy
- Marine

Platform

- Comm.
- Facility
- Helicopter
- Missile
- Plane
- Radar
- Satellite
- Ship

Contract Size

- Small
- Other

Appendix D: R Code TXT to CSV File

```
# http://jeffreybreen.wordpress.com/2011/07/04/twitter-text-mining-r-slides/

setwd("C:/txt file location")
sample = scan("1.txt", what=" ")
      # clean up samples with R's regex-driven global substitute, gsub():
       sample = gsub('[[:punct:]]', '', sample)
       sample = gsub('[[:cntrl:]]', '', sample)
       sample = gsub('\\d+', '', sample)
       # and convert to lower case:
       sample = tolower(sample)
sample1 <- as.data.frame(table(sample))
sample = scan("2.txt", what=" ")
      # clean up samples with R's regex-driven global substitute, gsub():
       sample = gsub('[[:punct:]]', '', sample)
       sample = gsub('[[:cntrl:]]', '', sample)
       sample = gsub('\\d+', '', sample)
       # and convert to lower case:
       sample = tolower(sample)
sample2 <- as.data.frame(table(sample))
sample = scan("3.txt", what=" ")
      # clean up samples with R's regex-driven global substitute, gsub():
       sample = gsub('[[:punct:]]', '', sample)
       sample = gsub('[[:cntrl:]]', '', sample)
       sample = gsub('\\d+', '', sample)
       # and convert to lower case:
       sample = tolower(sample)
sample3 <- as.data.frame(table(sample))
sample = scan("4.txt", what=" ")
      # clean up samples with R's regex-driven global substitute, gsub():
       sample = gsub('[[:punct:]]', '', sample)
       sample = gsub('[[:cntrl:]]', '', sample)
       sample = gsub('\\d+', '', sample)
       # and convert to lower case:
       sample = tolower(sample)
sample4 <- as.data.frame(table(sample))

wordcount <- merge(sample1,sample2,by="sample", all = TRUE)
wordcount <- merge(wordcount,sample3,by="sample", all = TRUE)
colnames(wordcount) <- c('sample','1','2','3')
wordcount <- merge(wordcount,sample4,by="sample", all = TRUE)
colnames(wordcount) <- c('sample','1','2','3','4')
wordcount[is.na(wordcount)] <- 0

setwd("C:/csv file location")
write.table(wordcount, file = "AEHF.csv", sep = ",", col.names = NA, qmethod = "double")
```

Appendix E: Excel VBA Code Remove Special Characters

```
Sub ReplaceInTextFile()
For I = 1 To 32
filelocation = "C:\TXT file location\" & I & ".txt"
Open filelocation For Input As #1
c0 = Input(LOF(1), #1)
Close #1

Open filelocation For Output As #1
Print #1, Replace(c0, """, "")
Close #1

Open filelocation For Input As #1
c0 = Input(LOF(1), #1)
Close #1

Open filelocation For Output As #1
Print #1, Replace(c0, """", "")
Close #1

Open filelocation For Input As #1
c0 = Input(LOF(1), #1)
Close #1

Open filelocation For Output As #1
Print #1, Replace(c0, " ", " ")
Close #1
Next I
End Sub
```

Appendix F: R Code Merge CSV Files

```
#save all program csv files in a common folder
#when merging csv files delete the first column (just numbers the variables) and change the column headings to program_# where # is the observation number

setwd("C:/file location with all csv file to be combined")

multmerge = function(mypath){
filenames=list.files(path=mypath, full.names=TRUE)
datalist = lapply(filenames, function(x){read.csv(file=x,header=T)})
Reduce(function(x,y) {merge(x,y, all = TRUE)}, datalist)}

wordcount = multmerge("C:/file location with all csv files to be combined")

wordcount[is.na(wordcount)] <- 0
write.table(wordcount, file = "wordcount.csv", sep = ",", col.names = NA, qmethod = "double")
```

Appendix G: Word VBA Code Extract Misspelled Words

```
Sub GetSpellingErrors()
'http://word.tips.net/T001465_Pulling_Out_Spelling_Errors.html
'1/10/2013
'Format 5s are professional documents. This implies that words should be spelled
‘accurately. If we find words that don't make sense we might be able to blame the
‘methods used for collecting the data from ‘PDF to txt files.
   Dim DocThis As Document
   Dim iErrorCnt As Integer
   Dim J As Integer

   Set DocThis = ActiveDocument
   Documents.Add

   iErrorCnt = DocThis.SpellingErrors.Count
   For J = 1 To iErrorCnt
      Selection.TypeText Text:=DocThis.SpellingErrors(J)
      Selection.TypeParagraph
   Next J
End Sub
```

Appendix H: Exempted Misspelled Words

Exempted Word	Explanation	Exempted Word	Explanation
eac	Estimate at Complete	timephasing	hyphens removed (and/or common usage)
sv	Schedule Variance	rephasing	hyphens removed (and/or common usage)
vac	Variance at Complete	supt	hyphens removed (and/or common usage)
wbs	Work Breakdown Structure	rqmts	hyphens removed (and/or common usage)
cpi	Cost Performance Index	underrunning	hyphens removed (and/or common usage)
bcwp	Budgeted Cost of Work Performed	timephased	hyphens removed (and/or common usage)
bcws	Budgeted cost of Work Scheduled	rephase	hyphens removed (and/or common usage)
acwp	Actual Cost of Work Performed	rephased	hyphens removed (and/or common usage)
spi	Schedule Performance Index	stopwork	hyphens removed (and/or common usage)
bac	Budget At Complete	underruning	hyphens removed (and/or common usage)
clin	Contract line item number	underran	hyphens removed (and/or common usage)
tcpi	Total Cost performance index	workpackage	hyphens removed (and/or common usage)
var	Variance	workpackages	hyphens removed (and/or common usage)
cpr	Contractor Performance Report	taskings	hyphens removed (and/or common usage)
obs	Obligations	underbudget	hyphens removed (and/or common usage)
underrun	hyphens removed (and/or common usage)	unscoped	hyphens removed (and/or common usage)
replan	hyphens removed (and/or common usage)	definitization	hyphens removed (and/or common usage)
pmb	program management baseline	definitized	hyphens removed (and/or common usage)

Exempted Word	Explanation	Exempted Word	Explanation
nonlabor	hyphens removed (and/or common usage)	replanned	hyphens removed (and/or common usage)
hrs	hyphens removed (and/or common usage)	lre	Latest Revised Estimate
mgmt	hyphens removed (and/or common usage)	underruns	hyphens removed (and/or common usage)
qual	hyphens removed (and/or common usage)	replanning	hyphens removed (and/or common usage)
unpriced	hyphens removed (and/or common usage)		

Appendix I: Definition 1: Naïve Bayes Classifier (LOOCV) Formulation

MI Threshold: 0.008

α- Level: 0.25

Naïve Bayes Text Classification Rule

$$c_{map} = \underset{c \in \mathbb{C}}{argmax} \left[\log \hat{P}(c) + \sum_{1 \le k \le n_d} \log \hat{P}(t_k|c) \right]$$

$\hat{P}(X_i = w|c_j)$

	$P(c_1)$	$P(c_2)$
	0.315164	0.684836
_Features	High-risk	Nominal Risk
absence	0.000183	0.000191
absences	5.97E-05	0.000131
ac	0.001001	0.001402
acct	3.08E-05	0.000259
accuracy	0.000132	0.000204
accurate	0.000849	0.000643
accurately	0.000219	0.000515
act	0.005998	0.002462
ad	0.000168	0.00053
addressed	0.00266	0.002183
adjusting	4.53E-05	0.000173
agreements	9.6E-05	0.000502
ahead	0.006266	0.009913
alerts	0.00011	3.26E-06
alternative	0.000161	0.000426
amp	5.25E-05	0.003127
amplifier	0.001501	0.000277
angular	0.000103	3.26E-06
anticipate	0.000755	0.001133
anticipation	0.000118	0.000293
aperture	0.000487	0.000139
apogee	8.15E-05	1.11E-05
applied	0.00208	0.00205
applying	0.000154	0.000353
apportioned	0.000306	0.000504
approvals	3.8E-05	0.000199
areas	0.004021	0.004002
assembly	0.015768	0.012798
assessed	0.001037	0.001123
assessments	0.000444	0.000588
assistance	0.000183	0.000249
assisting	8.87E-05	0.000131
attributable	0.001501	0.001517
attributed	0.00371	0.005516
audits	0.000378	0.000627
auto	0.000147	0.00023
avalanche	0.00019	5.87E-06
axis	0.000234	0.000199
basic	0.000784	0.000525
batteries	0.000277	0.000418
battery	0.00363	0.002444
beach	0.00027	0.000554
benefit	0.000364	0.000468
benefited	9.05E-06	0.000149
billed	7.42E-05	0.000507
billing	0.001226	0.001799
billings	8.87E-05	0.000345
bookcase	0.000197	1.11E-05

books	8.15E-05	0.000223	contingency	0.000125	0.000225
bounded	0.000154	2.15E-05	contributes	0.000226	0.000995
break	0.000378	0.001608	contributor	0.000719	0.001193
broken	0.00019	0.000737	coordinated	0.000349	0.000549
budgeting	3.08E-05	0.000264	coordination	0.001573	0.001624
build	0.018926	0.020628	cord	9.6E-05	3.26E-06
burst	0.000335	8.48E-06	correcting	0.000219	0.00022
business	0.00321	0.002099	critical	0.010648	0.011341
cage	0.000364	1.37E-05	da	0.00048	0.001193
cam	0.007092	0.009731	damaged	0.000922	0.000718
candidates	3.08E-05	0.000183	database	0.003536	0.003234
carrying	0.000118	0.000178	days	0.001964	0.003289
cat	0.000574	2.41E-05	deliveries	0.010155	0.008462
catching	0.000726	0.000815	deltas	4.53E-05	7.11E-05
cc	0.001124	0.003167	demonstrated	7.42E-05	0.000254
ceiling	0.001103	0.000392	demonstration	0.000922	0.000786
cell	0.001284	0.000959	desaturation	0.000103	3.26E-06
certifications	0.000168	0.000207	designing	3.08E-05	0.000149
changing	0.000494	0.000896	developmental	0.000147	0.000293
chassis	0.000552	0.000998	distributed	0.000574	0.000812
checkout	0.001631	0.002629	distribution	0.004956	0.003143
chillers	9.6E-05	5.87E-06	disturbance	0.000103	3.26E-06
claimed	0.001262	0.00213	diurnal	0.000154	3.26E-06
closeouts	0.000277	6.59E-05	diverting	0.000139	2.68E-05
closure	0.004992	0.003608	dos	0.000205	2.41E-05
coating	0.000118	0.000319	double	0.000277	0.000309
coded	0.00032	9.2E-05	downstream	0.000733	0.000429
codes	0.000248	0.000343	draft	0.000733	0.000888
coding	0.000234	0.00052	ds	0.000168	0.000215
combined	0.001914	0.001554	dynamic	0.000292	0.00034
compartment	0.000473	3.46E-05	early	0.009496	0.013072
compatible	0.000277	3.98E-05	earned	0.004731	0.008123
completions	0.000371	0.0004	effect	0.005274	0.005286
condensation	0.00011	1.37E-05	efficiency	0.004557	0.006064
conducting	0.000306	0.000562	elect	7.42E-05	0.000147
conference	3.8E-05	0.00017	email	0.000306	6.85E-05
configuration	0.004007	0.004576	enclosure	0.000458	0.001188
configurations	0.000719	0.000674	enhancements	0.001146	0.000303
cons	5.97E-05	1.11E-05	entire	0.000958	0.00088
consensus	0.000226	1.89E-05	equipment	0.008678	0.009616
considered	0.000263	0.000658	evaluated	0.001298	0.001608
consumption	3.08E-05	0.00023	evaluating	0.000386	0.000376

evolved	5.25E-05	7.37E-05	improved	0.001646	0.001835
executable	0.000103	1.11E-05	inability	0.000415	0.000549
exercise	0.000567	0.001039	include	0.006513	0.005534
expenditures	0.000929	0.0014	incorporations	6.7E-05	0.000152
faceplates	0.000125	3.26E-06	incorrect	0.000719	0.000849
factory	0.003283	0.002804	increases	0.002327	0.002504
failing	3.8E-05	0.000238	inductor	0.000154	3.26E-06
faraday	0.000364	1.37E-05	inefficiency	0.000378	0.000463
favorable	0.026987	0.036475	inexperienced	4.53E-05	0.000123
fc	0.000878	0.000632	integration	0.038408	0.031587
fi	5.25E-05	0.000384	integrator	0.000617	5.02E-05
finalizing	0.000444	0.000176	intensive	0.000197	0.000272
fire	0.000712	0.001045	intercostal	0.000349	1.89E-05
fitted	8.15E-05	3.26E-06	interferences	0.000313	9.98E-05
flight	0.022518	0.021152	intervention	0.000241	3.26E-06
floats	0.000292	1.11E-05	investigation	0.005383	0.003903
flushness	0.000263	8.48E-06	invoiced	0.000161	0.000319
forcing	2.35E-05	0.000136	ion	0.000849	0.000157
forecasting	0.00027	0.000444	isolation	0.000313	0.000724
fourth	6.7E-05	0.000246	items	0.008439	0.007178
frequent	0.000205	5.02E-05	keys	0.000313	8.16E-05
function	0.000683	0.000823	late	0.022323	0.018339
gauge	0.00019	4.24E-05	layout	0.00048	0.000671
gen	0.000292	0.000288	leakage	0.000154	0.000142
generates	9.05E-06	0.000131	lean	0.000197	0.000343
gimbals	0.000378	7.11E-05	lessen	0.00098	0.000134
global	0.001479	0.000457	lesson	0.000538	7.63E-05
golden	0.000205	0.000656	leveraging	0.000415	0.000494
government	0.007686	0.003532	liaison	0.000284	0.00028
greater	0.008193	0.006933	link	0.001515	0.001577
gusset	0.000197	5.87E-06	liquid	0.000161	5.02E-05
hardware	0.020135	0.016553	live	6.7E-05	0.000596
harnesses	0.00119	0.000883	loader	3.8E-05	0.000121
head	0.000748	0.001726	loaning	1.63E-05	0.000115
header	0.001226	0.000494	logistics	0.001501	0.002031
heads	0.001204	0.001133	logs	4.53E-05	0.000173
heritage	0.000241	8.94E-05	long	0.002595	0.003334
ho	0.000176	1.63E-05	los	0.000219	0.000303
house	0.000168	0.000418	losses	9.6E-05	0.000256
housings	0.000176	0.000288	macro	0.000132	4.24E-05
impacts	0.008845	0.008546	magnetics	0.000154	3.46E-05
implemented	0.002739	0.003318	making	0.000502	0.000557

mandate	0.000205	5.87E-06	performing	0.001993	0.002149
manpower	0.002508	0.002514	phasing	0.000552	0.000849
map	0.000378	0.000429	planed	0.000255	8.42E-05
media	0.001081	0.000178	planning	0.011532	0.009261
message	0.000465	6.59E-05	plans	0.006027	0.003877
messages	0.000241	3.72E-05	plating	0.000161	0.00057
metal	0.000342	0.00094	pointing	0.000914	0.000157
middle	0.000103	0.00017	port	0.001045	0.000316
minus	3.08E-05	0.000149	position	0.003326	0.003542
mitigation	0.011865	0.00383	predictability	9.05E-06	5.55E-05
modal	0.000675	2.41E-05	preparation	0.003898	0.00383
modem	0.001668	0.000384	preparations	0.000603	0.000776
modifications	0.003362	0.001452	preparing	0.000299	0.000465
motion	0.001334	0.000176	preserve	0.000335	6.59E-05
mount	0.00011	0.000209	previous	0.006969	0.005925
ne	0.002855	0.000431	primarily	0.019331	0.021463
newly	0.000335	0.000517	processes	0.002124	0.001708
nm	0.000502	0.000194	processing	0.004847	0.003806
notably	5.25E-05	1.11E-05	procurement	0.008627	0.006426
notching	0.000103	5.87E-06	proper	0.000335	0.00118
nulling	0.000313	1.11E-05	protocol	0.000596	0.000186
obsolescence	0.00048	0.000797	pubs	0.000371	0.000872
offset	0.010836	0.010117	purchasing	0.000632	0.000374
onetime	0.000234	0.00046	pure	1.63E-05	0.00011
op	8.15E-05	9.46E-05	quarter	0.001016	0.005552
opportunities	0.008656	0.007721	raised	0.000364	9.46E-05
opportunity	0.005274	0.002981	realized	0.007896	0.005184
optimally	0.000147	5.87E-06	reassembly	3.08E-05	0.000189
optimization	0.000263	0.000601	recalibration	4.53E-05	0.000126
ordered	0.000473	0.00081	recently	0.000567	0.00142
orientations	0.00011	3.26E-06	recovery	0.009866	0.016791
outsource	0.000683	0.001	redesigned	0.000219	0.000324
overly	9.05E-06	8.16E-05	reducing	0.001624	0.001444
oversee	0.000234	4.24E-05	ref	0.000958	0.000319
oversight	0.003616	0.001436	refine	0.000545	0.000144
overtime	0.00371	0.002556	relating	0.000212	0.000335
overview	0.000596	0.000204	reliability	0.000632	0.001099
page	0.0155	0.013466	relocation	0.001023	0.000429
panel	0.003579	0.002391	remain	0.001986	0.002673
parts	0.012227	0.010694	remainder	0.001313	0.001653
pedigree	0.000335	0.000269	removed	0.00124	0.001695
people	0.000378	0.000885	repair	0.003355	0.002812

repaired	0.000183	0.000264	staffed	0.000328	0.000609
repairs	0.000255	0.001071	staffing	0.012611	0.007638
rephase	0.000118	0.000228	stand	0.000393	0.000713
rephased	5.25E-05	0.000217	step	0.000386	0.000439
rephrased	1.63E-05	9.2E-05	stopping	0.000255	1.37E-05
requires	0.000798	0.000896	strengthen	0.000292	2.94E-05
rerouted	0.000103	3.26E-06	studies	0.002638	0.002227
research	0.00019	0.000538	subassembly	0.000132	0.000416
resumed	0.00011	0.000165	subtracted	1.63E-05	0.000144
return	0.001841	0.001355	subtracting	0.000197	2.94E-05
reworks	1.63E-05	0.000129	summaries	0.000328	0.000301
rod	3.08E-05	0.000126	supplies	0.00069	0.000293
rolling	0.001486	0.002715	surface	0.000161	0.000178
round	3.8E-05	0.000609	surge	0.000516	0.000742
route	3.08E-05	0.000118	survivability	0.000538	0.00046
runs	0.002529	0.003626	sustaining	0.00166	0.001517
samples	0.000147	6.33E-05	swirl	0.000103	3.26E-06
san	0.000342	0.000635	switchover	0.000205	1.11E-05
satellite	0.002138	0.000844	synergies	5.25E-05	0.000129
savings	0.002573	0.003824	sys	0.001733	0.00301
screen	0.000335	5.55E-05	tagging	0.000313	3.72E-05
scrub	0.00032	3.98E-05	taping	0.000574	8.48E-06
secondary	0.000154	0.000455	tasks	0.032288	0.041521
sets	0.000914	0.00142	tcpi	0.020715	0.023436
setups	0.000183	3.46E-05	tear	5.97E-05	0.000209
shared	0.000342	0.000379	technology	0.001153	0.000241
shelf	0.000299	6.33E-05	term	0.001544	0.001974
shielding	0.000762	0.000269	terminated	7.42E-05	0.000173
short	0.000806	0.001086	testers	2.35E-05	0.000246
significant	0.007186	0.010073	thermistor	0.000733	2.68E-05
simulation	0.001754	0.001992	thermo	3.8E-05	0.000238
sit	0.000241	0.000457	thin	0.000147	1.63E-05
slave	0.000103	5.87E-06	thousands	0.00187	0.003138
slipping	5.97E-05	0.000337	thrust	5.97E-05	0.000238
slosh	0.000103	3.26E-06	times	0.000965	0.001214
source	0.000987	0.001243	touchups	0.00011	3.26E-06
span	0.000197	0.000262	training	0.003666	0.005432
spare	0.001421	0.001525	transfer	0.003312	0.003255
spares	0.004564	0.004506	transition	0.002537	0.002639
special	0.001211	0.001781	trend	0.002616	0.003101
specifically	0.002529	0.002141	TRUE	0.000248	0.000481
spending	0.000552	0.001118	turnover	0.000255	0.000115

underruning	1.63E-05	0.000129	hand	0.000183	0.00022
underspend	3.8E-05	0.000115	identifying	0.000132	0.00028
unfavorable	0.036641	0.044392	incur	0.000212	0.000494
unknowns	0.000516	0.000463	incurring	0.000147	0.000319
unpriced	0.001957	0.001045	induction	5.25E-05	0.000358
unresolved	0.000277	4.76E-05	initiative	7.42E-05	0.00017
updating	0.000784	0.000708	managers	0.000393	0.000619
upgrade	0.005021	0.003707	materialize	4.53E-05	0.000134
users	0.000255	0.000533	methods	0.000241	0.000557
utilize	0.000567	0.000857	mixer	0.000226	6.33E-05
vac	0.07018	0.080204	offsite	0.000313	0.00041
valid	0.000168	0.000781	overstatement	8.15E-05	0.000178
validate	0.00053	0.000207	pegged	9.05E-06	0.000361
validity	7.42E-05	0.000162	physical	0.000357	0.000666
vectors	0.000118	1.11E-05	pools	0.000103	3.72E-05
venting	0.000125	3.26E-06	prism	0.000154	0.000643
verification	0.009308	0.00746	productive	6.7E-05	0.000155
wbs	0.05849	0.040364	recognize	5.25E-05	0.000256
weeks	0.001595	0.002441	representatives	0.000342	6.07E-05
wheel	0.000248	0.000168	respect	0.000299	0.000502
winglet	0.000139	3.26E-06	rest	0.000277	0.000439
wire	0.001407	0.003107	returning	0.00019	0.000319
wrong	5.97E-05	0.000379	rpm	0.00098	0.000144
yearend	1.63E-05	0.000162	slack	0.000588	0.00106
accelerate	0.000364	0.000437	specifications	0.000675	0.000502
accumulated	0.000407	0.000985	supportability	0.000755	0.000504
arrive	0.000429	0.000489	uncertainty	1.63E-05	8.68E-05
assemble	8.15E-05	0.000671	understatement	4.53E-05	0.000165
attained	3.08E-05	0.00028	adversely	9.6E-05	0.000233
bills	4.53E-05	0.000105	airframe	0.002442	0.002261
communicated	3.08E-05	0.000157	believed	0.000132	0.000186
coupling	3.08E-05	0.000209	big	9.05E-06	9.98E-05
defects	0.001501	0.000343	cad	0.000842	4.5E-05
deliverables	0.000444	0.000557	compounded	0.000241	2.15E-05
deviations	7.42E-05	0.000181	construction	0.001559	0.00111
diagrams	5.25E-05	0.000384	defining	9.6E-05	0.000376
directly	0.000552	0.000859	explain	3.08E-05	0.007298
directs	0.000139	5.29E-05	foundation	0.000147	1.37E-05
effectiveness	6.7E-05	0.000209	intercept	9.6E-05	1.89E-05
eleven	3.08E-05	7.63E-05	personal	2.35E-05	0.000147
entered	0.000473	0.000489	philosophy	0.000103	1.11E-05
forces	9.05E-06	0.000113	refinement	0.000415	0.000105

roles	3.08E-05	8.42E-05	alleviate	0.000205	8.68E-05
uhf	0.00208	0.000758	answer	0.00019	3.46E-05
ultra	9.6E-05	1.37E-05	artificially	5.25E-05	0.000115
vetted	0.00019	2.15E-05	attention	0.000118	0.000489
accomplishing	0.00011	0.00028	authoring	9.05E-06	0.00028
aircrew	9.05E-06	0.000236	aware	4.53E-05	0.000165
burdens	0.000219	6.07E-05	comparisons	1.63E-05	9.72E-05
compile	0.00019	1.37E-05	computing	8.15E-05	0.000228
diagram	0.00011	0.000504	context	0.000509	6.85E-05
disclosed	2.35E-05	0.000272	converting	9.05E-06	7.9E-05
familiarity	0.000125	1.89E-05	courseware	6.7E-05	0.000155
human	0.00027	0.000413	credits	3.8E-05	0.000225
obtaining	0.00011	0.000207	decline	3.8E-05	0.000136
published	0.000335	8.68E-05	developments	0.00027	7.37E-05
recouped	0.000147	1.63E-05	dim	1.81E-06	0.000413
skilled	0.000248	9.46E-05	dispositions	9.05E-06	7.9E-05
smiths	1.63E-05	0.000695	disruption	1.63E-05	0.000288
undergo	9.05E-06	9.46E-05	economies	9.05E-06	9.46E-05
unexpectedly	0.000168	2.68E-05	ensures	0.000255	3.26E-06
volatile	4.53E-05	5.87E-06	excessive	4.53E-05	0.000319
accumulation	0.000103	0.000225	fashion	8.87E-05	0.000288
aerodynamics	1.63E-05	0.000152	floating	2.35E-05	0.000332
calls	0.000168	0.000721	fulfill	9.05E-06	0.000173
consultants	1.63E-05	0.000275	gaps	3.08E-05	0.000283
cots	0.001472	0.001251	heavily	2.35E-05	0.000241
embedded	0.00019	0.000356	hourly	0.000951	0.001535
equates	5.25E-05	0.000173	inflated	4.53E-05	0.000118
hose	2.35E-05	0.00016	mil	6.7E-05	0.000236
hydraulic	0.000625	0.001246	missions	2.35E-05	0.000173
ids	0.000784	0.001217	node	5.97E-05	0.000262
indices	0.000183	0.001875	normalize	2.35E-05	0.000152
lighting	7.42E-05	0.000392	openings	6.7E-05	1.11E-05
likewise	5.25E-05	0.000126	overspent	9.05E-06	0.000152
modest	9.05E-06	0.000139	picked	1.63E-05	0.000118
obligations	5.97E-05	8.48E-06	pulls	0.000103	8.48E-06
predominantly	0.000219	0.000805	redevelop	0.000161	3.26E-06
stands	0.000241	0.000371	reflection	9.6E-05	0.000157
targets	0.000741	0.000408	segregated	0.000205	5.55E-05
tempo	3.08E-05	7.9E-05	simulators	0.000422	8.16E-05
ultimately	8.15E-05	0.000256	solutions	0.00237	0.000423
vacant	9.6E-05	0.000228	stage	0.001327	0.000857
web	0.000103	0.000249	strictly	9.05E-06	0.000228

suspension	8.87E-05	0.000228	restore	8.15E-05	1.63E-05
traveling	9.05E-06	0.00016	rogers	0.000777	0.00023
underestimating	0.000342	0.000426	shroud	4.53E-05	0.000168
unreleased	9.05E-06	7.11E-05	strategies	0.000429	8.16E-05
write	7.42E-05	0.000259	trended	8.15E-05	4.76E-05
agree	5.25E-05	8.48E-06	cutter	4.53E-05	5.87E-06
armor	6.7E-05	0.000457	depository	0.000132	0.000233
breaking	5.97E-05	1.63E-05	introduced	0.000132	0.00022
colocation	7.42E-05	0.000113	links	1.81E-06	0.000183
costly	5.25E-05	0.000173	mine	0.000248	0.00052
das	5.97E-05	0.000115	organized	0.00011	2.15E-05
drafting	0.000241	0.000468	outsourced	5.25E-05	0.000465
escalate	9.05E-06	0.000123	scaling	9.05E-06	0.000108
factoring	0.000103	2.41E-05	versions	0.000712	0.000243
hoses	1.63E-05	0.000215	blitz	0.000234	2.94E-05
invalid	2.35E-05	0.000105	deterioration	9.05E-06	0.000105
matured	9.6E-05	0.000173	faster	8.87E-05	0.000277
proving	9.05E-06	0.000131	sight	0.000139	0.000269
raw	0.000168	0.00058	svc	3.8E-05	7.63E-05
shafts	0.00011	0.000223	website	0.003246	0.00028
shot	9.05E-06	0.000113	productions	3.08E-05	7.9E-05
sizing	0.000284	2.41E-05	staring	9.6E-05	5.87E-06
slippages	5.97E-05	1.37E-05	administrator	0.000349	5.29E-05
stems	9.05E-06	8.94E-05	cps	0.000849	0.000361
sufficiently	7.42E-05	1.11E-05	deadline	0.000226	3.46E-05
brown	0.000168	1.63E-05	desktop	0.000183	0.000189
con	3.8E-05	0.000157	documenting	1.63E-05	0.000126
conventional	0.000103	1.37E-05	gyro	3.8E-05	0.000562
disassemble	9.05E-06	0.000152	individually	7.42E-05	1.11E-05
disassembly	0.000299	0.00047	rounds	9.05E-06	0.00022
draining	6.7E-05	1.11E-05	suffer	5.97E-05	3.26E-06
encounter	0.000168	2.94E-05	brigade	0.000161	1.37E-05
envisioned	1.63E-05	0.000259	broader	0.000103	1.89E-05
erection	7.42E-05	3.26E-06	captures	5.97E-05	0.000157
excavation	0.000118	1.11E-05	casino	0.000103	1.63E-05
feels	1.63E-05	9.98E-05	grand	0.000255	0.00011
foundations	8.87E-05	8.48E-06	hood	0.000139	1.63E-05
independently	0.000183	1.63E-05	hosting	1.63E-05	0.000126
linear	0.000125	0.00053	infantry	9.6E-05	8.48E-06
night	0.000132	1.37E-05	interactions	0.000292	3.2E-05
punching	6.7E-05	8.48E-06	managerial	0.000292	2.41E-05
resident	0.000292	9.46E-05	messaging	0.000125	5.87E-06

outset	0.000103	1.63E-05	stakeholder	8.15E-05	2.15E-05
pied	0.000248	4.5E-05	surfaces	2.35E-05	0.000118
spinout	0.000139	8.48E-06	unrealized	0.000154	3.72E-05
uncontrolled	0.003239	0.000277	chances	8.87E-05	1.63E-05
virtually	0.00011	3.2E-05	compartments	9.6E-05	5.87E-06
wad	0.001081	0.000277	hydro	0.000132	1.37E-05
maritime	0.000407	3.72E-05	outfitting	0.000205	1.63E-05
pie	0.00245	0.00046	scalable	6.7E-05	1.89E-05
seeker	0.000422	0.001637	swath	0.000176	4.76E-05
unacceptable	0.000219	5.02E-05	unfamiliar	3.8E-05	0.000233
devoting	0.000132	5.87E-06	documentations	5.97E-05	1.89E-05
tips	0.000197	5.55E-05	choke	9.6E-05	2.41E-05
accessible	0.000726	9.46E-05	connect	0.000125	2.15E-05
brad	0.000328	1.37E-05	enveloped	7.42E-05	1.89E-05
heavier	0.000241	0.000157	fax	0.00019	3.98E-05
instructed	0.000349	3.72E-05	formation	5.25E-05	3.26E-06
refactoring	0.000125	1.63E-05	forums	0.000241	2.41E-05
advantages	8.87E-05	5.87E-06	gore	6.7E-05	1.89E-05
boundary	6.7E-05	0.000293	helices	0.000212	2.68E-05
coop	0.000103	1.11E-05	helix	0.000393	1.63E-05
encompassing	7.42E-05	1.37E-05	leveled	9.05E-06	6.59E-05
fields	5.25E-05	1.89E-05	multiband	0.000103	1.11E-05
interconnect	0.000168	0.000345	rebuilds	1.63E-05	0.000223
interconnection	7.42E-05	2.94E-05	synthesizers	8.87E-05	8.48E-06
predictions	2.35E-05	0.000217	flood	0.000378	3.98E-05
routers	0.000219	8.42E-05	mater	8.15E-05	5.87E-06
screens	0.000473	5.81E-05	protector	0.000313	2.15E-05
spaces	0.000118	3.72E-05	scintillation	9.6E-05	1.89E-05
struggle	0.000255	3.98E-05	semesters	9.6E-05	1.11E-05
today	9.05E-06	0.000123	turnovers	0.000118	1.11E-05
arrowhead	6.7E-05	1.11E-05	uniformly	0.000132	1.37E-05
emitter	0.000125	3.2E-05	viability	0.000139	1.11E-05
malfunctions	7.42E-05	5.87E-06	wizards	0.000132	1.63E-05
modeled	1.63E-05	8.42E-05	facet	9.05E-06	7.9E-05
proximity	9.05E-06	0.00029	sibs	0.000139	3.2E-05
raise	0.000168	2.41E-05	harnessing	9.05E-06	8.16E-05

Appendix J: Definition 2: Hybrid Classifier (LOOCV) Formulation

Hybrid Model (Part I: Naïve Bayes classifier to produce outputs for Part II)

MI Threshold: 0.007

α- Level: 0.00006103515625

Naïve Bayes Text Classification Rule

$$c_{map} = \underset{c \in \mathbb{C}}{argmax} \left[\log \hat{P}(c) + \sum_{1 \le k \le n_d} \log \hat{P}(t_k|c) \right]$$

	$P(c_1)$	$P(c_2)$
	0.27881	0.72119
$\hat{P}(X_i = w\|c_j)$		
_Features	High-Risk	Nominal-Risk
absence	0.000109	0.000255
absences	1.36E-05	0.000197
ac	0.001425	0.002137
acct	5.43E-05	0.00035
accurately	0.000339	0.000659
act	0.009542	0.003473
activation	0.000665	0.000148
ad	8.14E-05	0.000762
adjusting	5.43E-05	0.000269
administration	0.003461	0.00151
advance	0.00133	0.000614
agreements	0.000163	0.00065
ahead	0.008809	0.01433
alerts	0.000122	4.48E-06
allowable	2.71E-05	0.000197
alternative	0.000217	0.00056
amp	4.07E-05	0.004884
amplifier	0.002158	0.00043
angular	0.000109	4.48E-06
anticipation	0.000136	0.000453
aperture	0.000624	0.000197
apogee	0.000136	8.96E-06
appendix	0.000516	0.000143
apportioned	0.000285	0.000766
approvals	4.07E-05	0.000287
areas	0.00566	0.005628
assessed	0.001493	0.001492
assessments	0.000489	0.000816
assistance	0.000122	0.000358
attributable	0.001765	0.002182
auxiliary	9.5E-05	0.000609
availed	8.14E-05	4.48E-06
avalanche	0.000204	8.96E-06
avoid	0.000842	0.000408
axis	0.000176	0.000278
base	0.006637	0.004122
baseplate	6.79E-05	4.48E-06
basic	0.001208	0.000807
benefit	0.000434	0.000663
benefited	1.36E-05	0.000202
billed	8.14E-05	0.000609
billings	0.000136	0.00043
bonds	0.000204	1.79E-05
bookcase	0.000244	4.48E-06
bounded	0.00019	2.69E-05
break	0.000584	0.002379
broken	0.000163	0.001026
budgeting	2.71E-05	0.000376

buffer	0.000299	5.83E-05	count	0.00114	0.000497
burst	0.000176	1.34E-05	cutout	8.14E-05	4.48E-06
cage	0.000421	1.79E-05	da	2.71E-05	0.001515
cam	0.008551	0.012743	damaged	0.000991	0.000932
campaign	5.43E-05	0.003383	demonstrated	0.000122	0.000345
candidates	1.36E-05	0.000278	derived	0.001059	0.001152
carry	0.001357	0.000614	desaturation	0.000109	4.48E-06
cat	0.000638	4.03E-05	designing	4.07E-05	0.000193
catching	0.001181	0.001134	detector	0.00038	0.000139
cc	0.001439	0.004311	develop	0.004628	0.002841
ceiling	0.001479	0.000533	directed	0.003353	0.001523
cell	0.001493	0.001349	discovered	0.002009	0.001344
certifications	0.000231	0.000291	discretely	1.36E-05	0.000161
changing	0.00076	0.001326	distributed	0.000611	0.001205
checkout	0.001642	0.003522	disturbance	0.000109	4.48E-06
chillers	0.000109	8.96E-06	diurnal	0.000122	4.48E-06
claimed	0.001575	0.00272	diverting	0.000176	2.24E-05
click	0.000217	4.48E-06	dos	0.000299	3.58E-05
closeouts	0.000231	8.96E-05	downstream	0.001167	0.000627
closure	0.006271	0.004906	drawers	9.5E-05	4.48E-06
cm	0.019695	0.005942	drops	0.002701	0.000641
coded	0.000516	0.000134	ds	9.5E-05	0.0003
coding	0.000312	0.000672	early	0.012569	0.019075
combined	0.002606	0.002151	earned	0.006407	0.01053
compartment	0.000434	5.83E-05	economical	0.000109	1.79E-05
compatible	0.000394	5.38E-05	eddy	0.00057	4.03E-05
condensation	0.000109	1.34E-05	edge	0.000692	0.000166
conducting	0.000407	0.000802	elect	1.36E-05	0.000175
conference	2.71E-05	0.000229	email	0.000421	9.41E-05
configuration	0.005361	0.006493	enclosure	0.000597	0.001443
configurations	0.00095	0.000999	enhance	0.000244	5.83E-05
cons	0.000109	8.96E-06	enhancements	0.001493	0.000444
consensus	0.000326	2.69E-05	equipment	0.011415	0.013411
considered	0.000258	0.000977	evaluated	0.001914	0.00216
consult	6.79E-05	4.48E-06	exercise	0.000679	0.001515
consumed	0.000204	0.000636	expenditures	0.001303	0.001895
consumption	4.07E-05	0.000314	faceplates	0.000109	4.48E-06
containers	0.000557	0.000851	failing	6.79E-05	0.000345
contingency	0.000109	0.000332	faraday	0.000421	1.79E-05
contributes	0.000366	0.001416	favorable	0.035128	0.049997
contributor	0.001072	0.001756	fi	2.71E-05	0.000609
coordinated	0.000516	0.000739	finalizing	0.000624	0.000242
cord	0.000109	4.48E-06	findings	0.000271	0.000632

fire	0.000747	0.001573	lessen	0.001371	0.000255
fitted	9.5E-05	4.48E-06	lesson	0.000774	0.000125
flags	0.000176	1.79E-05	liquid	0.000271	5.38E-05
flatness	0.000543	0.000108	live	5.43E-05	0.000869
floats	0.000312	1.79E-05	loader	6.79E-05	0.000179
flushness	0.000271	1.34E-05	loaning	1.36E-05	0.000175
forecasting	0.000366	0.000654	logistics	0.002321	0.002706
fourth	0.000122	0.000345	logs	6.79E-05	0.000233
frequent	0.000326	7.17E-05	long	0.003081	0.005036
function	0.000882	0.001143	loop	0.001466	0.00026
gasket	0.000136	2.24E-05	los	0.000122	0.000421
gauge	0.000231	5.83E-05	losses	0.000122	0.000345
gave	0.000109	1.34E-05	macro	0.000204	4.48E-05
gen	0.000394	0.000381	made	0.006841	0.008823
generates	1.36E-05	0.000211	magnetics	0.000204	4.93E-05
gimbals	0.000624	0.000112	maintained	0.00114	0.000385
global	0.00243	0.000659	mandate	0.000217	8.96E-06
grounding	0.001303	0.000363	media	0.001344	0.00026
gusset	0.00019	8.96E-06	members	0.000597	0.001259
header	0.001927	0.000686	message	0.000733	8.51E-05
heritage	0.000366	0.000134	messages	0.000353	7.17E-05
ho	0.000258	2.24E-05	minus	5.43E-05	0.000197
house	0.000122	0.000618	mitigate	0.007886	0.0044
identifiers	0.003434	0.000385	mm	0.000176	0.000493
inability	0.000312	0.000802	modal	0.000665	4.03E-05
incorporations	8.14E-05	0.000229	modem	0.001194	0.000542
increases	0.002945	0.003625	motion	0.001493	0.000273
inductor	0.000204	4.48E-06	motors	2.71E-05	0.000305
inductors	0.000109	4.48E-06	nadir	0.000557	7.62E-05
inefficiency	0.000624	0.000587	ne	0.004045	0.000551
inexperienced	4.07E-05	0.000175	negotiated	0.004791	0.003746
integration	0.051823	0.043378	newly	0.000461	0.000712
integrator	0.000747	7.17E-05	notching	0.000109	8.96E-06
intercostal	0.000394	2.69E-05	nulling	0.000136	1.79E-05
intervention	0.000109	4.48E-06	obsolescence	0.000624	0.001044
investigation	0.004819	0.005108	offset	0.014944	0.013514
invoiced	0.000163	0.000457	onetime	0.000312	0.000686
ion	0.001127	0.000184	opportunity	0.007723	0.004033
isolation	0.000434	0.001049	optimally	0.000176	8.96E-06
iv	0.000787	0.001371	optimization	0.000312	0.000896
keys	0.000407	0.000103	organization	0.001018	0.001125
layout	0.000557	0.000977	orientations	0.000163	4.48E-06
lean	0.000204	0.000551	outsource	0.000909	0.001447

overly	1.36E-05	0.000108	remain	0.00243	0.003885
oversee	0.000326	5.38E-05	remainder	0.001534	0.002402
oversight	0.004751	0.0022	removals	0.000258	1.34E-05
overtime	0.004574	0.003647	removed	0.001344	0.002429
overview	0.000828	0.000323	repaired	0.000136	0.00035
page	0.022165	0.01851	repairs	0.000339	0.00147
park	0.000258	8.96E-06	reporting	0.018962	0.017722
partners	0.000394	7.62E-05	rerouted	9.5E-05	4.48E-06
people	0.000461	0.001264	research	0.000285	0.000784
performing	0.002457	0.002904	retain	0.000244	8.07E-05
phased	0.001357	0.000789	reworks	1.36E-05	0.000188
phases	0.002145	0.00091	rocket	2.71E-05	0.000184
phasing	0.000638	0.00112	rolling	0.001832	0.003638
plating	0.00019	0.000793	roughly	0.000353	0.000703
plugs	0.000122	1.79E-05	round	3.31E-09	0.000878
pointing	0.001181	0.000237	route	2.71E-05	0.00017
pop	0.005171	0.001071	samples	0.000271	4.48E-05
port	0.001642	0.00043	san	0.000285	0.000954
position	0.005076	0.004839	satellite	0.003244	0.001264
preparations	0.000557	0.001035	saver	6.79E-05	4.48E-06
preparing	0.000407	0.00065	savings	0.003746	0.005243
presentation	9.5E-05	0.000515	screen	0.000543	8.07E-05
preserve	0.000543	0.000108	scrub	0.000502	4.48E-05
problems	0.00433	0.004732	secondary	0.000258	0.000641
processes	0.00319	0.002433	senor	9.5E-05	8.96E-06
processing	0.005592	0.005435	setups	0.000299	4.93E-05
procurement	0.011103	0.009051	shared	0.000285	0.00052
proper	0.000461	0.001725	shelf	0.000502	9.86E-05
protocol	0.000842	0.000269	shielding	0.000787	0.00039
pubs	0.000434	0.001362	short	0.000991	0.00151
pure	2.71E-05	0.000161	shunting	6.79E-05	4.48E-06
quarter	0.00167	0.007971	simulation	0.001398	0.002962
raised	0.000502	0.000152	sit	0.000217	0.000672
ratio	0.002592	0.001828	size	0.002389	0.000968
reason	0.005864	0.006793	slave	9.5E-05	8.96E-06
reassembly	5.43E-05	0.000255	slipping	8.14E-05	0.000421
recessed	8.14E-05	4.48E-06	slosh	0.000109	4.48E-06
recovering	0.000611	0.000968	solution	0.00243	0.00086
reducing	0.001927	0.001909	span	0.000149	0.00035
ref	0.001289	0.000466	specifically	0.003923	0.002989
refine	0.000774	0.000188	spending	0.000774	0.001618
relating	0.000258	0.00047	spread	0.000869	0.001429
reliability	0.000665	0.001523	staffing	0.016234	0.010705

stand	0.000448	0.00091	unscheduled	0.000692	0.000341
static	0.002511	0.001026	updating	0.000842	0.001004
step	0.000516	0.000596	upgrade	0.007737	0.004929
stockroom	0.000176	1.34E-05	utilizes	0.000163	3.14E-05
strengthen	0.000448	3.58E-05	vac	0.095094	0.107673
studies	0.00243	0.003195	valid	0.000231	0.000986
subassembly	0.000176	0.000614	validate	0.000652	0.000287
subtracting	0.000271	4.48E-05	validity	4.07E-05	0.000206
supplies	0.001059	0.000372	vectors	0.000136	4.48E-06
surface	9.5E-05	0.000264	venting	0.000136	4.48E-06
surge	0.000543	0.001044	voiding	6.79E-05	4.48E-06
survivability	0.000543	0.000672	wave	0.002131	0.003419
swirl	0.000109	4.48E-06	wbs	0.082743	0.057981
switchover	0.000217	1.79E-05	weekend	0.000448	0.000166
sys	0.001724	0.004113	weeks	0.001955	0.003482
table	0.00323	0.001963	west	0.000516	0.000224
tagging	0.000475	6.27E-05	winglet	0.000109	4.48E-06
taping	0.000597	1.34E-05	worked	0.004398	0.004669
tasks	0.042362	0.058084	wrong	8.14E-05	0.000515
tcpi	0.025762	0.031065	xl	0.000258	7.62E-05
tear	9.5E-05	0.000296	yearend	2.71E-05	0.000202
term	0.0019	0.002944	arrive	0.000584	0.000659
testers	2.71E-05	0.00035	assemble	0.000122	0.000945
thermistor	0.00076	4.48E-05	aviation	0.004357	0.001411
thermo	2.71E-05	0.000341	cease	2.71E-05	0.000143
thin	0.000149	2.24E-05	communicated	2.71E-05	0.000224
times	0.00133	0.001734	defects	0.002348	0.000497
tolerance	0.000204	0.000278	deliverables	0.000638	0.000771
touchups	0.000109	4.48E-06	departments	0.000285	8.96E-05
traceability	0.000638	0.000161	deviations	6.79E-05	0.00026
tracking	0.004086	0.001649	diagrams	9.5E-05	0.000551
training	0.005212	0.007658	diminish	0.000109	0.000255
transition	0.002905	0.003777	directly	0.000611	0.00125
trend	0.002362	0.005001	disassembled	1.36E-05	0.000188
TRUE	0.000339	0.00069	effectiveness	4.07E-05	0.000282
tubes	0.00057	0.000358	eleven	1.36E-05	0.000125
turnover	0.000366	0.00017	forces	1.36E-05	0.000152
unavailability	0.001194	0.000488	forecasts	0.000448	0.000807
underruning	2.71E-05	0.000161	identifying	0.000149	0.000408
underspend	1.36E-05	0.000188	incur	0.000271	0.000708
unfavorable	0.049637	0.061342	incurring	0.000217	0.000466
unpriced	0.002606	0.001479	initiative	2.71E-05	0.000278
unresolved	0.000475	5.38E-05	managers	0.000516	0.000883

methods	0.000312	0.000793	hazard	5.43E-05	0.000229
mixer	0.000407	9.41E-05	indexes	0.000217	5.83E-05
noted	0.002402	0.001313	negligible	2.71E-05	0.000587
pegged	1.36E-05	0.000538	obtaining	8.14E-05	0.000309
percentages	0.000149	1.34E-05	published	0.000516	0.000134
pools	0.000149	5.83E-05	recouped	0.000231	3.58E-05
prism	6.79E-05	0.001031	schematic	0.000231	0.000139
recognize	9.5E-05	0.000381	skilled	0.000421	0.000117
representatives	0.000543	8.96E-05	slope	9.5E-05	8.96E-06
respect	0.00038	0.000672	smiths	2.71E-05	0.001134
rest	0.000271	0.000654	terms	0.000299	0.000502
rpm	0.00171	0.000193	unexpectedly	0.000285	1.34E-05
sow	0.003706	0.001792	accumulation	0.000122	0.0003
uncertainty	1.36E-05	0.000103	calls	0.000231	0.001026
adversely	0.000122	0.000354	consultants	1.36E-05	0.000399
asked	2.71E-05	0.000161	cots	0.001914	0.001739
cad	0.001452	8.96E-05	embedded	0.000258	0.000453
comply	0.000176	0.000506	equates	1.36E-05	0.000246
compounded	0.000394	3.14E-05	handle	0.000869	0.000372
construction	0.001968	0.0016	helped	5.43E-05	0.000291
defining	9.5E-05	0.000484	hose	4.07E-05	0.000215
describe	2.71E-05	0.002021	hydraulic	0.000923	0.001703
ended	0.000774	0.000341	ids	0.001072	0.001676
explain	5.43E-05	0.011171	indices	0.000285	0.00242
foundation	0.000244	3.14E-05	lighting	8.14E-05	0.000578
governments	0.000461	8.07E-05	likewise	1.36E-05	0.000193
intercept	0.000163	2.69E-05	modest	1.36E-05	0.000175
keeping	0.000109	0.000435	narratives	4.07E-05	0.000305
misinterpretation	0.00038	0.00013	obligations	0.000109	8.96E-06
parameter	0.000149	1.79E-05	predominantly	0.000339	0.001066
personal	4.07E-05	0.000206	reserved	0.000217	5.83E-05
philosophy	0.000163	1.79E-05	targets	0.001181	0.000533
programmatic	0.000366	0.002684	trip	0.006855	0.004216
refinement	0.000557	0.000157	ultimately	9.5E-05	0.000358
roles	2.71E-05	0.00013	understaffed	0.000149	0.000421
vetted	0.000326	3.14E-05	vacant	1.36E-05	0.000314
accomplishing	0.000176	0.000394	web	0.000122	0.00035
aircrew	1.36E-05	0.000358	yielding	6.79E-05	0.000211
burdens	0.000366	9.41E-05	adherence	0.000502	5.83E-05
compile	0.000285	2.24E-05	alleviate	0.000299	0.000125
diagram	0.000149	0.000793	answer	0.000299	5.38E-05
disclosed	2.71E-05	0.000367	attention	0.000176	0.000721
familiarity	0.000204	2.69E-05	authoring	1.36E-05	0.000417

aware	6.79E-05	0.000246	traveling	1.36E-05	0.000224
causal	1.36E-05	0.000175	underestimating	0.000584	0.000596
comparisons	1.36E-05	0.000152	unreleased	1.36E-05	0.000108
computing	6.79E-05	0.000327	write	8.14E-05	0.000363
consist	4.07E-05	0.000193	armor	9.5E-05	0.000686
context	0.000814	6.72E-05	augmented	2.71E-05	0.000237
converting	1.36E-05	0.000112	costly	3.31E-09	0.000282
decline	5.43E-05	0.000193	das	5.43E-05	0.000148
developments	0.000394	0.000108	entrance	0.000217	4.48E-05
dim	3.31E-09	0.000632	escalate	1.36E-05	0.000166
dispositions	1.36E-05	0.000125	factoring	0.00019	1.34E-05
disruption	2.71E-05	0.000444	hoses	2.71E-05	0.0003
economies	1.36E-05	0.000125	invalid	2.71E-05	0.000139
ensures	0.000475	4.48E-06	prescribed	6.79E-05	4.48E-06
excessive	5.43E-05	0.000439	proving	1.36E-05	0.000202
extends	0.000529	8.07E-05	raw	0.000271	0.000802
fashion	0.000136	0.000412	redesigning	0.000122	1.79E-05
floating	4.07E-05	0.000412	shafts	0.000176	0.000314
fulfill	1.36E-05	0.00026	shot	1.36E-05	0.000157
gaps	5.43E-05	0.000412	sizing	0.000475	4.03E-05
gaskets	1.36E-05	0.000139	steel	0.000176	0.000605
heavily	1.36E-05	0.000314	stems	1.36E-05	0.000125
hires	1.36E-05	0.000273	sufficiently	0.000136	1.34E-05
hourly	0.001371	0.002191	theory	0.000977	0.000242
inflated	4.07E-05	0.000166	uncovered	0.000529	0.000193
leader	1.36E-05	0.000143	backfilling	0.000149	8.96E-06
mil	8.14E-05	0.000345	brown	0.000285	2.69E-05
missions	2.71E-05	0.000251	chemical	0.003556	0.002491
node	5.43E-05	0.000381	con	4.07E-05	0.000193
normalize	2.71E-05	0.000188	disassembly	5.43E-05	0.000798
omitted	0.000271	6.72E-05	draining	0.000122	8.96E-06
overspent	1.36E-05	0.000211	encounter	0.000285	4.48E-05
picked	1.36E-05	0.000179	envisioned	3.31E-09	0.000354
pulls	0.00019	8.96E-06	excavation	0.000217	1.79E-05
redevelop	0.000244	4.48E-06	feels	2.71E-05	0.000139
redline	0.000176	0.000448	independently	0.000312	2.24E-05
reflection	0.000122	0.000211	linear	4.07E-05	0.000789
scanning	1.36E-05	0.000161	mast	0.000869	0.000139
shipside	2.71E-05	0.000215	night	0.00019	1.79E-05
simulators	0.000638	0.000125	pervious	0.000204	1.34E-05
solutions	0.003	0.000529	platforms	0.000461	0.001232
strictly	1.36E-05	0.000332	pumps	0.000176	2.24E-05
suspension	1.36E-05	0.000363	punching	0.000122	4.48E-06

recruiting	1.36E-05	0.000125	halted	2.71E-05	0.000134
resident	0.000394	0.000134	hood	0.000258	2.69E-05
restore	0.000136	2.69E-05	hosting	2.71E-05	0.000179
shroud	2.71E-05	0.000264	infantry	0.000149	1.34E-05
strategies	0.000652	0.000108	interactions	0.000475	4.03E-05
substation	0.000624	1.34E-05	managerial	0.000339	2.69E-05
wage	0.000176	0.002429	messaging	0.000231	8.96E-06
cutter	8.14E-05	4.48E-06	nick	9.5E-05	8.96E-06
depository	3.31E-09	0.000345	outset	0.000163	2.24E-05
exceptions	1.36E-05	0.000108	pied	0.000394	7.17E-05
interruption	2.71E-05	0.000148	reproduce	0.000258	2.69E-05
links	3.31E-09	0.000242	spin	0.000407	0.000229
organized	0.000163	1.79E-05	spinout	0.000231	1.34E-05
outsourced	6.79E-05	0.000677	terrestrial	0.000285	4.48E-06
scaling	1.36E-05	0.000152	theater	0.000529	9.41E-05
versions	0.000991	0.000323	uncontrolled	0.004995	0.000394
voice	0.000597	0.000197	pie	0.00357	0.000453
algorithmic	0.00019	3.14E-05	producible	1.36E-05	0.000117
blitz	0.00038	4.93E-05	sib	0.001914	0.000202
deterioration	1.36E-05	0.000157	sources	0.000271	6.27E-05
largo	0.002022	0.00039	unacceptable	0.000353	6.72E-05
med	0.000176	0.000426	devoting	0.000163	8.96E-06
sight	1.36E-05	0.000394	shape	0.000217	2.69E-05
spiral	0.000977	0.000103	tips	0.000326	1.79E-05
svc	2.71E-05	0.000112	accessible	0.001018	0.000117
website	0.004995	0.000399	brad	0.000407	2.24E-05
productions	2.71E-05	0.000103	heavier	0.000326	0.000215
staring	0.000163	8.96E-06	instructed	0.00057	4.93E-05
administrator	0.000611	7.62E-05	java	0.000258	4.48E-05
cps	0.001222	0.000453	refactoring	0.00019	2.69E-05
deadline	0.000394	4.03E-05	strike	0.000271	5.83E-05
desktop	0.000271	0.00026	touches	1.36E-05	9.86E-05
documenting	2.71E-05	0.000175	advantages	0.000136	8.96E-06
gyro	6.79E-05	0.000811	boundary	0.000122	0.000412
mobility	2.71E-05	0.000215	collectively	0.000122	1.79E-05
preservation	0.000109	4.48E-06	coop	0.000163	1.79E-05
rounds	1.36E-05	0.000305	fields	9.5E-05	4.48E-06
suffer	8.14E-05	4.48E-06	highlight	8.14E-05	8.96E-06
brigade	0.000271	2.24E-05	interconnect	0.00019	0.000475
broader	0.000163	2.69E-05	interconnection	0.000136	4.48E-05
captures	4.07E-05	0.00022	label	0.000529	0.000152
casino	0.000163	2.24E-05	lighter	2.71E-05	0.000488
cci	0.000136	4.48E-06	predictions	2.71E-05	0.000296

routers	0.000366	0.00013	connect	0.000204	2.69E-05
screens	0.000801	9.41E-05	downsized	9.5E-05	4.48E-06
sop	0.000271	3.58E-05	enveloped	0.000136	2.24E-05
spaces	0.000204	4.93E-05	fax	0.000271	4.03E-05
struggle	0.000353	6.72E-05	formation	8.14E-05	4.48E-06
today	1.36E-05	0.000175	forums	0.000394	3.58E-05
arrowhead	0.000122	1.34E-05	frequencies	0.000163	4.48E-06
editor	1.36E-05	9.41E-05	helix	0.000733	8.96E-06
emitter	0.000204	5.38E-05	multiband	0.000176	1.79E-05
malfunctions	0.000122	4.48E-06	rebuilds	2.71E-05	0.000318
modeled	2.71E-05	0.000117	synthesizers	0.000163	4.48E-06
proximity	1.36E-05	0.000417	equations	0.000312	8.96E-05
raise	0.000244	4.03E-05	flood	0.000638	6.72E-05
surfaces	1.36E-05	0.000179	mater	0.000136	8.96E-06
unrealized	0.000258	4.03E-05	orb	0.000217	4.48E-05
auxiliaries	0.000434	2.69E-05	protector	0.000543	3.58E-05
chances	0.000122	2.24E-05	scintillation	0.000163	2.69E-05
compartments	0.000136	8.96E-06	semesters	0.000149	1.79E-05
hookups	0.000149	4.48E-06	turnovers	0.00019	1.79E-05
hydro	0.00019	2.24E-05	uniformly	0.000217	2.24E-05
maneuvering	0.000163	8.96E-06	viability	0.000244	8.96E-06
outfitting	0.000312	2.69E-05	wizards	0.000217	2.69E-05
conduction	0.000122	4.48E-05	facet	1.36E-05	0.000112
suitcase	6.79E-05	4.48E-06	sibs	0.000258	5.38E-05
documentations	0.000109	1.34E-05	sleeper	0.000122	8.96E-06
choke	0.000176	3.14E-05	harnessing	1.36E-05	0.000112

Hybrid Model (Part II: Using inputs from Naïve Bayes Classifier above)

Variables:

TSPI
CV%
NB_Pred_Class

$$R_1: -\frac{1}{2}\boldsymbol{x_0'}(\mathbf{S}_1^{-1} - \mathbf{S}_2^{-1})\boldsymbol{x_0} + (\boldsymbol{\mu}_1'\mathbf{S}_1^{-1} - \boldsymbol{\mu}_2'\mathbf{S}_2^{-1})\boldsymbol{x_0} - k \geq \ln\left[\left(\frac{c(1|2)}{c(2|1)}\right)\left(\frac{p_2}{p_1}\right)\right]$$

$$R_2: -\frac{1}{2}\boldsymbol{x_0'}(\mathbf{S}_1^{-1} - \mathbf{S}_2^{-1})\boldsymbol{x_0} + (\boldsymbol{\mu}_1'\mathbf{S}_1^{-1} - \boldsymbol{\mu}_2'\mathbf{S}_2^{-1})\boldsymbol{x_0} - k < \ln\left[\left(\frac{c(1|2)}{c(2|1)}\right)\left(\frac{p_2}{p_1}\right)\right]$$

Where

$$\text{k} = \frac{1}{2}\ln\left(\frac{|\boldsymbol{S_1}|}{|\boldsymbol{S_2}|}\right) + \frac{1}{2}(\boldsymbol{\mu}_1'\mathbf{S}_1^{-1}\boldsymbol{\mu_1} - \boldsymbol{\mu}_2'\mathbf{S}_2^{-1}\boldsymbol{\mu_2})$$

Let

$$\ln\left[\left(\frac{c(1|2)}{c(2|1)}\right)\left(\frac{p_2}{p_1}\right)\right] = 0.9372$$

$$\boldsymbol{S_1} = \begin{pmatrix} 1.497529 & -0.03483 & 0.056895 \\ -0.03483 & 0.005637 & -0.0064 \\ 0.056895 & -0.0064 & 0.213246 \end{pmatrix}$$

$$\boldsymbol{S_2} = \begin{pmatrix} 2.052543568 & -0.03787 & 0.125756 \\ -0.037865621 & 0.006774 & -0.00631 \\ 0.125756345 & -0.00631 & 0.0140152 \end{pmatrix}$$

$$\boldsymbol{\mu}_1 = \begin{pmatrix} 1.30106685 \\ -0.0495351 \\ 0.69611307 \end{pmatrix}$$

$$\boldsymbol{\mu}_2 = \begin{pmatrix} 1.240413 \\ -0.01239 \\ 0.168508 \end{pmatrix}$$

Appendix K: Definition 3: Multivariate Classification (LOOCV) Formulation

Variables:

- % Complete
- CPI
- % Difference Between ML and W
- % Difference Between W and B
- TCPI StDev
- SCI StDev
- CV% StDev
- AF
- Comm.
- Helicopter
- Ship
- Plane
- Radar
- Satellite
- Small

$$R_1: -\frac{1}{2}\boldsymbol{x_0'}(\mathbf{S_1^{-1}} - \mathbf{S_2^{-1}})\boldsymbol{x_0} + (\boldsymbol{\mu_1'}\mathbf{S_1^{-1}} - \boldsymbol{\mu_2'}\mathbf{S_2^{-1}})\boldsymbol{x_0} - k \geq \ln\left[\left(\frac{c(1|2)}{c(2|1)}\right)\left(\frac{p_2}{p_1}\right)\right]$$

$$R_2: -\frac{1}{2}\boldsymbol{x_0'}(\mathbf{S_1^{-1}} - \mathbf{S_2^{-1}})\boldsymbol{x_0} + (\boldsymbol{\mu_1'}\mathbf{S_1^{-1}} - \boldsymbol{\mu_2'}\mathbf{S_2^{-1}})\boldsymbol{x_0} - k < \ln\left[\left(\frac{c(1|2)}{c(2|1)}\right)\left(\frac{p_2}{p_1}\right)\right]$$

Where

$$\mathrm{k} = \frac{1}{2}\ln\left(\frac{|\boldsymbol{S_1}|}{|\boldsymbol{S_2}|}\right) + \frac{1}{2}(\boldsymbol{\mu_1'}\mathbf{S_1^{-1}}\boldsymbol{\mu_1} - \boldsymbol{\mu_2'}\mathbf{S_2^{-1}}\boldsymbol{\mu_2})$$

Let

$$\ln\left[\left(\frac{c(1|2)}{c(2|1)}\right)\left(\frac{p_2}{p_1}\right)\right] = -0.06375$$

$S_1 =$

0.04325977	-0.00386	-0.00142	-0.00222	0.013355	-0.00265	-0.00068	-0.0059	-0.01514	-0.00321	0.006534	-0.01747	0.009259	0.021156	0.011196
-0.003857411	0.004196	-0.00111	-0.00072	0.002747	0.00043	-0.0002	0.003241	-0.00182	0.001335	-0.00142	0.004141	0.004076	-0.00636	-0.0001
-0.001422716	-0.00111	0.001511	0.001462	-0.00111	0.000163	0.000253	0.000946	-0.00027	-7.9E-05	-0.00015	0.002056	0.000221	-0.00153	0.00302
-0.002222975	-0.00072	0.001462	0.002111	0.002957	0.000235	0.000276	0.001922	-0.00051	0.001085	-0.00076	0.000385	0.002004	-0.00202	0.000309
0.013355093	0.002747	-0.00111	0.002957	0.749532	0.000308	0.001046	0.024825	-0.04351	-0.01619	0.002125	-0.03069	0.061541	0.02933	0.005883
-0.002646645	0.00043	0.000163	0.000235	0.000308	0.001611	0.000962	0.000155	-0.00031	-0.00058	-0.00022	0.00175	-0.00081	0.000211	0.003509
-0.000675908	-0.0002	0.000253	0.000276	0.001046	0.000962	0.000965	-0.00101	-0.00071	-0.00025	-4.9E-05	-0.0003	-0.00047	0.001837	0.002405
-0.005900557	0.003241	0.000946	0.001922	0.024825	0.000155	-0.00101	0.221683	-0.10534	-0.0084	-0.01887	0.033203	0.060604	0.045085	-0.07569
-0.015135087	-0.00182	-0.00027	-0.00051	-0.04351	-0.00031	-0.00071	-0.10534	0.217498	-0.03334	-0.01819	-0.06669	-0.0288	-0.06442	0.000605
-0.003212807	0.001335	-7.9E-05	0.001085	-0.01619	-0.00058	-0.00025	-0.0084	-0.03334	0.093813	-0.00597	-0.0219	-0.00946	-0.02115	0.035771
0.006533837	-0.00142	-0.00015	-0.00076	0.002125	-0.00022	-4.9E-05	-0.01887	-0.01819	-0.00597	0.053885	-0.01194	-0.00516	-0.01154	-0.01382
-0.0174699	0.004141	0.002056	0.000385	-0.03069	0.00175	-0.0003	0.033203	-0.06669	-0.0219	-0.01194	0.165728	-0.01891	-0.0423	0.014399
0.009258853	0.004076	0.000221	0.002004	0.061541	-0.00081	-0.00047	0.060604	-0.0288	-0.00946	-0.00516	-0.01891	0.08231	-0.01827	-0.03696
0.021156049	-0.00636	-0.00153	-0.00202	0.02933	0.000211	0.001837	0.045085	-0.06442	-0.02115	-0.01154	-0.0423	-0.01827	0.16152	0.007793
0.011195909	-0.0001	0.00302	0.000309	0.005883	0.003509	0.002405	-0.07569	0.000605	0.035771	-0.01382	0.014399	-0.03696	0.007793	0.242212

$S_2 =$

0.05296623	-0.00754	-0.00342	-0.00462	0.134185	-0.00209	-0.00062	0.035553	-0.03436	-0.0069	-0.00061	0.030267	0.002471	0.014324	-0.0262
-0.007543212	0.007864	0.000983	0.00243	-0.05658	0.000663	1.39E-05	-0.00294	-0.00142	-0.00113	-0.00048	-0.00211	-0.00087	-0.00415	0.008972
-0.003421111	0.000983	0.001287	0.001404	-0.00737	4.73E-05	-3.4E-06	-0.00312	0.006498	0.000185	-0.00042	-0.00312	0.001035	-0.00161	0.006662
-0.004624765	0.00243	0.001404	0.0021	0.000309	0.00015	-6E-06	-0.00242	0.005121	0.001015	-0.00074	-0.0034	0.000768	-0.00104	0.007865
0.134184643	-0.05658	-0.00737	0.000309	61.61797	0.014849	0.021547	-0.30168	-0.171	-0.03663	-0.03544	-0.25333	-0.01444	0.646754	0.675922
-0.002085373	0.000663	4.73E-05	0.00015	0.014849	0.000566	0.000238	0.000293	-0.0013	0.000325	-0.00032	-0.00073	-6.1E-05	-0.00011	0.001678
-0.00061533	1.39E-05	-3.4E-06	-6E-06	0.021547	0.000238	0.000166	0.00014	-0.00051	0.000248	-0.00015	-0.00096	-5.6E-06	0.000564	0.001257
0.035552585	-0.00294	-0.00312	-0.00242	-0.30168	0.000293	0.00014	0.232243	-0.0731	0.004575	-0.0148	0.119244	0.016128	0.035192	-0.05694
-0.034362335	-0.00142	0.006498	0.005121	-0.171	-0.0013	-0.00051	-0.0731	0.160406	-0.00863	-0.00812	-0.0599	-0.00508	-0.02386	0.054315
-0.006896094	-0.00113	0.000185	0.001015	-0.03663	0.000325	0.000248	0.004575	-0.00863	0.04129	-0.00175	-0.01289	-0.00109	-0.00513	0.014322
-0.000609865	-0.00048	-0.00042	-0.00074	-0.03544	-0.00032	-0.00015	-0.0148	-0.00812	-0.00175	0.038964	-0.01213	-0.00103	-0.00483	-0.00802
0.030267378	-0.00211	-0.00312	-0.0034	-0.25333	-0.00073	-0.00096	0.119244	-0.0599	-0.01289	-0.01213	0.210024	-0.00758	-0.03564	-0.04899
0.002471214	-0.00087	0.001035	0.000768	-0.01444	-6.1E-05	-5.6E-06	0.016128	-0.00508	-0.00109	-0.00103	-0.00758	0.024738	-0.00302	-0.00501
0.0143244	-0.00415	-0.00161	-0.00104	0.646754	-0.00011	0.000564	0.035192	-0.02386	-0.00513	-0.00483	-0.03564	-0.00302	0.105095	0.017053
-0.026202756	0.008972	0.006662	0.007865	0.675922	0.001678	0.001257	-0.05694	0.054315	0.014322	-0.00802	-0.04899	-0.00501	0.017053	0.158877

$$\boldsymbol{\mu}_1 = \begin{bmatrix} 0.71229952 \\ 0.95996639 \\ 0.02984478 \\ 0.04094059 \\ 0.18126513 \\ 0.01702859 \\ 0.01097231 \\ 0.33016627 \\ 0.31828979 \\ 0.10451306 \\ 0.05700713 \\ 0.20902613 \\ 0.09026128 \\ 0.20190024 \\ 0.40855107 \end{bmatrix}$$

$$\boldsymbol{\mu}_2 = \begin{bmatrix} 0.63897418 \\ 1.00982778 \\ 0.02195576 \\ 0.03119706 \\ 0.87479956 \\ 0.01059470 \\ 0.00556278 \\ 0.36455696 \\ 0.20000000 \\ 0.04303797 \\ 0.04050633 \\ 0.29873418 \\ 0.02531646 \\ 0.11898734 \\ 0.19746835 \end{bmatrix}$$

Bibliography

Air Force Cost Analysis Agency. (2007). Earned Value Managment. In *Air Force Cost Analysis Handbook* (pp. 13.1-13.78).

Calcutt, H. M. (1994). *Cost Growth in DoD Major Programs: A Historical Perspective.* Washington, D.C.: The Industrial College of the Armed Forces.

Defense Acquisition University. (2009, December 22). *ACQupedia*. Retrieved July 08, 2012, from Defense Acquisition University: https://acc.dau.mil/CommunityBrowser.aspx?id=189574

Defense Acquisition University. (2012, November 6). *11.3.1.4 Contractor Performance Management Reporting*. Retrieved December 19, 2012, from Defense Acquisition Guidebook: https://acc.dau.mil/CommunityBrowser.aspx?id=488729#11.3.1.4.1

Defense Cost and Resource Center. (2005, March 30). *OSD.mil*. Retrieved July 31, 2012, from http://dcarc.pae.osd.mil/Files/EVMCR/CPR_DID.pdf

Defense Cost and Resource Center. (2013a). *DCARC Portal*. Retrieved January 25, 2013, from CPR File Viewer User Guide: http://dcarc.cape.osd.mil/Files/EVMCR/CPR_File_Viewer_User_Guide.pdf

Defense Cost and Resource Center. (2013b). *EVM-CR Dashboard*. Retrieved February 24, 2013, from DCARC: https://service.dcarc.cape.osd.mil/EVM/Site/DashBoards/EVMDash.aspx

Department of Defense. (2011). *SAR Cost Variance Instructions.* Washington: GPO.

Department of Defense. (2012a). *Overview - FY 2013 Defense Budget.* Washington: GPO.

Department of Defense. (2012b, January). *U.S. Department of Defense.* Retrieved July 2, 2012, from Defense Budget Priorites and Choices: http://www.defense.gov/news/Defense_Budget_Priorities.pdf

Department of Defense. (2012c). *IPMR Implementation Guide.* Washington: GPO.

Department of the Air Force. (2009). *Guide to Acquisition and Sustainment Life Cycle Management.* Washington: GPO.

Doak, J. (1992). *An Evaluation of Feature Selection Methods and Their Application to Computer Security.*

Dowling, A. W. (2012, March). Using Predictive Analytics to Detect Major Problems in Department of Defense Acquisition Programs. *(Accession No. ADA557925).* Defense Technical Information Center.

Dowling, A. W., Miller, T. P., & White, E. (2012). Problem Detection for DoD Acquisition Programs. Alexandria: Military Operations Research Symposium.

Forman, G. (2008). Feature Selection for Text Classification. In H. Liu, & H. Motoda, *Computational Methods of Feature Selection* (pp. 257-276). Boca Raton: Chapman & Hall/CRC.

Hough, P. G. (1992). *Pitfalls in Calculating Cost Growth from Selected Acquisition Reports.* Santa Monica: RAND.

Jennrich, R. I. (1977a). Stepwise Regression. In K. Enslein, A. Ralston, & H. S. Wilf, *Statistical Methods for Digital Computers Vol III* (pp. 58-64). New York: Wiley-Interscience.

Jennrich, R. I. (1977b). Stepwise Discriminant Analysis. In K. Enslein, A. Ralston, & H. S. Wilf, *Statistical Methods for Digital Computers* (pp. 76-79). New York: Wiley-Interscience.

Johnson, R. A., & Wichern, D. W. (2007). *Applied Multivariate Statistical Analysis 6th ed.* Upper Saddle River: Pearson Education, Inc.

Keaton, C. G., White, E. D., & Unger, E. J. (2011). Using Earned Value Data to Detect Potential Problems in Acquisition Contracts. *Journal of Cost Analysis and Parametrics, Volume 4*(Issue 2), 148-159.

Kutner, M. H., Nachtsheim, C. J., Neter, J., & Li, W. (2005). *Applied Linear Statistical Models.* New York: McGraw-Hill/Irwin.

Kwak, Y. H. (2012). History, Practices, and Future of Earned Value Management in Government: Perspectives from NASA. *Project Mnaagement Journal*, 77-90.

Liu, H., & Motoda, H. (2008). *Computational Methods of Feature Selection.* Boca Raton: Chapman & Hall/CRC.

Manning, C. D., Raghavan, P., & Schutze, H. (2008). *Introductino to Information Retrieval.* New York: Cambridge University Press.

Microsoft. (2010a). Excel. (Version 14). Redmond, WA.

Microsoft. (2010b). Word. (Version: 14.06129.500 (32-bit)). Redmond, WA, United States of America.

Miller, T. (2012, March). Acquisition Program Problem Detection Using Text Mining Methods. *(Accession No. ADA557568)*. Defense Technical Information Center.

Office of the Under Secretary of Defense (Comptroller). (2011, March). *National Defense Budget Estimates for FY 2012.* Retrieved July 2, 2012, from Department of Defense: http://comptroller.defense.gov/defbudget/fy2012/FY12_Green_Book.pdf

OUSD(AT&L). (2006). *Risk Management Guide for DOD Acquistion.* Washington: GPO.

RAND. (2008). *Sources of Weapon System Cost Growth.* Santa Monica: RAND Corporation.

Salton, G. (1971). *The SMART Retrival System.* Englewood Cliffs: Prentice-Hall, Inc.

SAS Institute INC. (2013a). JMP. (Version 9). Cary, NC.

SAS Institute Inc. (2013b). *Multivariate Details*. Retrieved Feburary 20, 2013, from JMP Statistical Discovery from SAS: http://www.jmp.com/support/help/Multivariate_Details.shtml

SAS Institute Inc. (2013c). *Statistical Details for the Distribution Platform*. Retrieved January 24, 2013, from JMP Support: http://www.jmp.com/support/help/Statistical_Details_for_the_Distribution_Platform.shtml

Sullivan, Michael J. (2001, March 29). *Director Acquisition and Sourcing Management.* Washington: GPO.

The R Foundation for Statistical Computing. (2011, September 30). R. Retrieved from R-Project: http://www.r-project.org/

Thompson, J. R., & Koronacki, J. (2002). *Statistical Process Control: The Deming Paradigm and Beyond.* Boca Raton: Chapman & Hall.

United States Government Accountability Office. (2012). *Assessments of Selected Weapon Programs.* Washington: GPO.

White, E. D., Sipple, V. P., & Greiner, M. A. (2004). Using Logistic and Multiple Regression to Estimate Engineering Cost Risk. *The Journal of Cost Analysis and Management*, 67-79.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From – To)*
21-03-2013	Master's Thesis	22 Aug 2011 – 21 Mar 2013

4. TITLE AND SUBTITLE
Multivariate and Naïve Bayes Text Classification Approach to Cost Growth Risk in Department of Defense Acquisition Programs

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Freeman, Charlton E., Captain, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)
Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/ENV)
2950 Hobson Way, Building 640
WPAFB OH 45433-8865

8. PERFORMING ORGANIZATION REPORT NUMBER
AFIT-ENC-13-M-03

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Intentionally Left Blank

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
DISTRIBUTION STATEMENT A: APPROVED FOR RELEASE; DISTRIBUTION IS UNLIMITED

13. SUPPLEMENTARY NOTES

14. ABSTRACT

In these fiscally austere times, researchers have diligently sought methods to detect cost risk in the DOD acquisition programs. Our research effort reflects a culmination of three years of research seeking solutions to the problem of identifying programs with elevated levels of cost risk. Specifically, we applied multivariate classification and multinomial Naïve Bayes text classification techniques to develop three cost risk identification models. We find our model considering a 6-month change in the estimate at complete (EAC) of greater than 5% in magnitude, identified 69.5% of the high-risk programs in our dataset with 76.21% accuracy. Next, our model considering a 6-month increase in the EAC of greater than 5% correctly identified 67.90% of the high-risk programs with 79.68% accuracy. Finally, our model considering a 12-month increase in the EAC of greater than 5%, identified 91.69% of the high-risk programs with an accuracy of 78.31%. This research effort acts as a capstone, concentrating the knowledge collected from previous efforts and provides an actionable decision support tool for the DOD acquisition community. We find this research directly supports the goals of "more disciplined use of resources" and "improving efficiency" laid out in the OUSD(Comptroller) FY2013 Defense Budget (Department of Defense, 2012a:3.1).

15. SUBJECT TERMS
Text Analysis, Cost Growth, Risk Analysis, Risk Classification

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			White, Edward D, Ph.D
U	U	U	UU	174	19b. TELEPHONE NUMBER *(Include area code)* (937) 255-3636, x 4540 (edward.white@afit.edu)